PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : G07F 7/10

A2

(11) International Publication Number: **WO 98/52163**

(43) International Publication Date: 19 November 1998 (19.11.98)

(21) International Application Number: PCT/GB98/01405

(22) International Filing Date: 14 May 1998 (14.05.98)

(30) Priority Data:

60/046,514	15 May 1997 (15.05.97)	US
09/075,973	11 May 1998 (11.05.98)	US

(71) Applicant: MONDEX INTERNATIONAL LIMITED [GB/GB]; 47–53 Cannon Street, London EC4M 5SQ (GB).

(72) Inventors: RICHARDS, Timothy, Philip; 32 Craig Mount, Radlett, Herts. WD7 7LW (GB). EVERETT, David, Barrington; 31 Ashdown Avenue, Saltdean, Brighton, Easr Sussex BN2 8AH (GB). VINER, John, Charles; Hydes, Woodlands Lane, Windlesham (GB).

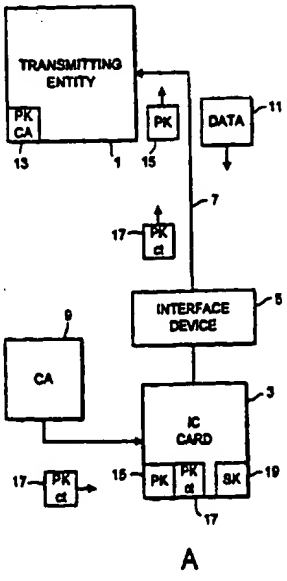
(74) Agent: POTTER, Julian, Mark; D. Young & Co., 21 New Fetter Lane, London EC4A 1DA (GB).

(81) Designated States: AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GE, GH, GM, GW, HU, ID, IL, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, UZ, VN, YU, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG).

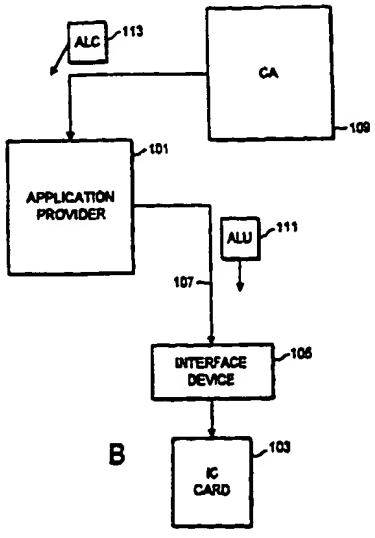
Published

Without international search report and to be republished upon receipt of that report.

(54) Title: IC CARD TRANSPORTATION KEY SET



A



B

(57) Abstract

Method and apparatus for securely transporting data onto an IC card. The method is used, for example, to transport data, including application programs, in a secure manner from a source located outside the IC card. At least a portion of the data is encrypted using the public key of a public/secret key pair of the intended IC card unit. The encrypted data is then sent to the IC card and the IC card verifies the key transformation unit using its unique secret key. The data can then be stored on the IC card. A copy of the public key signed by a certification authority can be used to verify that the card is authorized to be part of the overall authorized system.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece			TR	Turkey
BG	Bulgaria	HU	Hungary	ML	Mali	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MN	Mongolia	UA	Ukraine
BR	Brazil	IL	Israel	MR	Mauritania	UG	Uganda
BY	Belarus	IS	Iceland	MW	Malawi	US	United States of America
CA	Canada	IT	Italy	MX	Mexico	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NE	Niger	VN	Viet Nam
CG	Congo	KE	Kenya	NL	Netherlands	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NO	Norway	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's Republic of Korea	NZ	New Zealand		
CM	Cameroon			PL	Poland		
CN	China	KR	Republic of Korea	PT	Portugal		
CU	Cuba	KZ	Kazakstan	RO	Romania		
CZ	Czech Republic	LC	Saint Lucia	RU	Russian Federation		
DE	Germany	LI	Liechtenstein	SD	Sudan		
DK	Denmark	LK	Sri Lanka	SE	Sweden		
EE	Estonia	LR	Liberia	SG	Singapore		

IC CARD TRANSPORTATION KEY SET

BACKGROUND OF INVENTION

Integrated circuit ("IC") cards are becoming increasingly used for many different purposes in the world today. An IC card (also called a smart card) typically is the size of a conventional credit card which contains a computer chip including a microprocessor, read-only-memory (ROM), electrically erasable programmable read-only-memory (EEPROM), an Input/Output (I/O) mechanism and other circuitry to support the microprocessor in its operations. An IC card may contain a single application or may contain multiple independent applications in its memory. MULTOS™ is a multiple application operating system which runs on IC cards, among other platforms, and allows multiple applications to be executed on the card itself. This allows a card user to run many programs stored in the card (for example, credit/debit, electronic money/purse and/or loyalty applications) irrespective of the type of terminal (i.e., ATM, telephone and/or POS) in which the card is inserted for use.

A conventional single application IC card, such as a telephone card or an electronic cash card, is loaded with a single application when it is manufactured and before it is given to a card user. That application, however, cannot be modified or changed after the card is issued even if the modification is desired by the card user or card issuer. Moreover, if a card user wanted a variety of application functions to be performed by IC cards issued to him or her, such as both an electronic purse and a credit/debit function, the card user would be required to carry multiple physical cards on his or her person, which would be quite

cumbersome and inconvenient. If an application developer or card user desired two different applications to interact or exchange data with each other, such as a purse application interacting with a frequent flyer loyalty application, the card user would be forced to swap multiple cards in and out of the card-receiving terminal, making the transaction difficult, lengthy and inconvenient.

Therefore, it is beneficial to store multiple applications on the same IC card. For example, a card user may have both a purse application and a credit/debit application on the same card so that the user could select which type of payment (by electronic cash or credit card) to use to make a purchase. Multiple applications could be provided to an IC card if sufficient memory exists and an operating system capable of supporting multiple applications is present on the card. Although multiple applications could be preselected and placed in the memory of the card during its production stage, it would also be beneficial to have the ability to load and delete applications for the card post-production as needed.

The increased flexibility and power of storing multiple applications on a single card create new technical challenges to be overcome concerning the integrity and security of the information (including application code and associated data) exchanged between the individual card and the application provider as well as within the entire system when loading and deleting applications. It would be beneficial to have the capability in the IC card system to exchange data among cards, card issuers, system operators and application providers securely and to load and delete applications securely at any time from a local terminal or remotely over a telephone line, Internet or intranet connection or other data conduit. Because

these data transmission lines are not typically secure lines, a number of security and entity authentication techniques must be implemented to make sure that applications being sent over the transmission lines are not tampered with and are only loaded on the intended cards.

As mentioned, it is important -- particularly where there is a continuing wide availability of new applications to the cardholder -- that the system has the capability of adding applications onto the IC card subsequent to issuance. This is necessary to protect the longevity of the IC cards; otherwise, once an application becomes outdated, the card would be useless. It would be beneficial to allow the addition of applications from a remote location as well as from a direct connection to an application provider's terminal. For example, it would be beneficial for a card user to be able to plug his or her IC card into a home computer and download an application over the Internet. This type of remote loading of applications raises a number of security risks when transmitting the application code and related data over an unsecured communications line such as the Internet.

An entity which transmits an application or data to an IC card requires that only the intended IC card should receive the transmitted data. Third parties should not be able to intercept and view the data. Additionally, a transmitting entity will require verification that the IC card which has requested information is actually part of the overall IC card system and not simply posing as being part of the system. These concerns are raised by both remote application loading as well as local terminal application loading.

Accordingly, it is an object of embodiments of this invention to provide a transfer technique having improved security and specifically to provide an IC-card system that allows for the transfer of data with improved security including smart card applications which may be loaded onto IC cards.

SUMMARY OF THE INVENTION

These and other objectives are achieved by an embodiment of the present invention which provides an IC card method and apparatus for securely transporting data including an application onto an IC card including storing a secret and public key pair on the IC card. retrieving the stored public key from the IC card, encrypting at least a portion of the data to be transported using the public key, transmitting the encrypted data to the IC card and decrypting the encrypted data using the IC card's secret key.

In a preferred embodiment. a certification authority ("CA") or the entity that manages the overall security of the IC card system. encrypts (or digitally signs) a copy of the IC card's public key and the signed copy is also stored on the IC card. The entity transmitting the data to the IC card can verify that the CA has approved the card by retrieving using the IC card's signed public key and verifying the signed public key using the public key of the CA. If verification is successful, the entity has verified that the CA approved the IC card.

BRIEF DESCRIPTION OF THE DRAWINGS

Further objects, features and advantages of embodiments of the invention will become apparent from the following detailed description taken by way of example only in conjunction with the accompanying figures showing illustrative embodiments of the invention, in which

Fig. 1A is a block diagram of the secure data transfer system which securely transfers data from a transferring entity to an IC card.

Fig. 1B is block diagram of the application loading system which loads an application from an application provider to an IC card;

Fig. 2 is a graphic representation of the contents of an Application Loading Unit;

Fig. 3 is a graphic representation of an Application Unit;

Fig. 4 is a flow chart of the steps for providing an individual key set for an IC card;

Fig. 5 is a graphic representation of a Key Transformation Unit;

Fig. 6 is a graphic representation of a Key Transformation Unit plaintext;

Fig. 7 is a graphic representation of the Application Load Certificate;

Fig. 8 is a graphic representation of the Application Unit being decrypted;

Fig. 9 is a flowchart illustrating the steps undertaken in processing the Application Load Unit;

Fig. 10 is a flowchart illustrating the steps undertaken in processing the KTU; and

Fig. 11 is a block diagram showing the components of an IC card which can receive and process and Application Load Unit.

Throughout the figures, the same reference numerals and characters, unless otherwise stated, are used to denote like features, elements, components or portions of the illustrated embodiments. Moreover, while the subject invention will now be described in detail with reference to the figures, it is done so in connection with the illustrative embodiments. It is intended that changes and modifications can be made to the described embodiments without departing from the true scope and spirit of the subject invention as defined by the appended claims.

DETAILED DESCRIPTION OF THE INVENTION

It is beneficial to have the capability to load applications onto IC cards containing multiple application operating systems at any time during the lifetime of the IC card. This flexibility allows a user of a card to periodically add new applications to the IC card and also allows older applications to be updated with newer versions of the application when they are released. For example, a card user may start with an IC card that contains a purse, or electronic cash application (e.g., MONDEX™), being stored on his IC card. Some time after the user has the card, he or she may load an additional application onto the card such as a credit/debit application. Some time after loading the credit/debit application on the

card, a new version of the credit/debit application may become available and the card user should be able to erase the old application on his IC card and replace it with the new version of the credit/debit application which may contain additional features. Additionally, an IC card needs to receive data regarding personal information such as new credit card account numbers or updated information.

The flexibility of loading applications and transmitting data at different times during the IC card's life cycle creates security issues with the process of loading applications onto the card. In a multiple application operating system environment, it is beneficial to be able to load applications and data both at terminals, such as a bank ATM machine, as well as over remote communication links, such as telephone lines, cable lines, the Internet, satellite or other communications means. When loading applications and data onto an IC card, the application provider needs to provide security regarding the applications to be loaded. First, the application provider must make sure the application is only sent to the correct card user who is intended to receive the application. Second, the application and associated data may contain private or trade secret information which needs to be encrypted so entities other than the IC card cannot view the contents of the encrypted application code and data. A portion of the application code and data may be secret while other portions are not. These concerns of authentication and protecting the contents of some or all of the application and associated data being loaded onto a card is addressed herein.

A number of encryption/decryption techniques are described herein. There are two basic types of encryption, symmetric encryption and asymmetric

encryption. Symmetric encryption uses a secret key as part of a mathematical formula which encrypts data by transforming the data using the formula and key. After the data is encrypted, another party can decrypt the encrypted data using the same secret key with a decryption algorithm. Thus the same key is used for encryption and decryption so the technique is symmetric. A conventional example of a symmetric algorithm is DES.

Asymmetric encryption techniques use two different keys of a pair for encrypting and decrypting information. The two keys are normally referred to as a private or secret key and a public key. When data is encrypted with one key of the pair, the other key is used to decrypt the data. If a sender of data signs the data with his secret key, anyone with the public key can verify the message. Since public keys are typically known to the public, the contents of a data signed with a secret key cannot be protected but the origination of the data can be verified by determining if a particular secret key signed the data. This authentication process is termed a digital signature. If person A wanted to authenticate a message he was sending to person B, the person A would sign the document with his secret key. When person B received the message, he would use person A's public key to verify the message. If the message was verified with the public key, person B would know that the document was signed with secret key of person A. Thus, the origin of the message has been authenticated.

The asymmetric key set can also be used to protect the contents of a message. If person A wanted to send an encrypted message to person B that no one else could read, he would encrypt the data or message with person B's public key

and send it to person B. Now only the holder of B's secret key could decrypt the data. If a combination of keys is used, a person could both authenticate and encrypt the message. The asymmetric pair of keys has some powerful applications with respect to card security. However, asymmetric encryption is relatively processor costly (processor cost is associated with computation time) compared with symmetric encryption. An example of asymmetric encryption method is RSA®.

A hybrid of symmetric encryption which makes the encryption method more powerful is to encrypt data using two symmetric keys. This technique is called triple DES which encodes data with key 1, decodes the data using key 2 (which in effect further encodes the data) and then further encodes the data using key 1 again. Once the data has arrived at its destination, key 1 is used to decode the data, key 2 is used to encode the data, and key 1 is used to decode the data. These extra steps of encoding and decoding make the technique more powerful and more difficult to properly decipher without both keys.

Figure 1A shows a block diagram of the entities used in transporting data in a secure manner in an IC card system. The transmitting entity 1 can be a card issuer, bank, IC card or other entity which desires to transport data to an IC card 3. The transmitting entity 1 preferably initiates the data transfer process. Alternatively, the IC card 3 can initiate the data transfer process if the card requires data from the transmitting entity 1.

The transmitting entity 1 is connected to interface device 5 (e.g., a terminal that communicates with an IC card). Data conduit 7 can be a telephone line, an intranet, the Internet, a satellite link or any other type of communications

link. In this example, the transmitting entity 1, which is remotely located from IC card 3, desires to send data in a secure manner to the IC card. However, because the data link is an "open" link (i.e. not a private link) and subject to third parties possibly intercepting or replacing data being transmitted, security measures are needed to guarantee that only the intended IC card will receive the transmitted data. The Certificate Authority 9 can also be used to authenticate that the IC card has been validated as part of the IC card system.

In Figure 1A, a private (or secret) key 19 and corresponding public key 15 is generated for IC card 3. The keys are preferably generated using an asymmetric encryption algorithm such as RSA®. The keys can be generated at the CA 9 or any other location because they are specific only to the IC card 3 and no other copies need to be kept. A third data item, the public key certificate 17, is also generated and stored on the IC card 3.

The public key certificate 17 is generated by signing the public key 15 with the private key of the CA 9. This allows a person with the public key of the CA 9 to verify that the CA digitally signed the IC card's public key in order to certify the IC card's individual key set. The public key certificate can be generated by the CA at the time the IC card private/public key set is generated or at a subsequent time.

When a data transfer is initiated by the transmitting entity 1, the IC card 3 is contacted through the interface device 5 and the IC card 3 sends its public key 15 and its public key certificate 17 to the transmitting entity 1. The transmitting entity then verifies the public key certificate with public key of the CA

13 (which is publicly available from the CA 9 and may be stored in the transmitting entity 1) thus determining if the CA 9 digitally signed the public key and verifying that the IC card is a valid card.

The transmitting entity 1 then encrypts the data to be transmitted with the IC card's public key. The transmitting entity 1 then transmits the encrypted data 11 to the interface device 5 and to the IC card 3. The IC card 3 decrypts the encrypted data with its corresponding private (also called secret) key 19. The data can then be processed by the IC card 3. Only the IC card 3 has a copy of its private key so only the intended IC card can access the encrypted data. This ensures that third parties cannot access the encrypted data and correspondingly that only the intended IC card will be able to read and process the data.

Figure 1B shows a secure method for loading applications onto an IC card. Figure 1B shows a block diagram of the entities used in a secure remote application loading process. The application provider 101 can be a card issuer, bank or other entity which provides application loading services. The application provider 101 initiates an application loading process onto IC card 103. IC card 103 is connected to data conduit 107 which is connected to interface device 105 (e.g., a terminal that communicates with an IC card). Data conduit 107 can be a telephone line, an intranet, the Internet, a satellite link or any other type of communications link. The application provider 101, which is remotely located from the IC card 103, desires to send and load an application to the IC card. However, because the data link is an open link and subject to third parties possibly intercepting or replacing applications being transmitted, security measures which authenticate the

application itself, the application provider and the IC card must be used to ensure the integrity of the system. The CA 109 may also be used to help authenticate that some data being transferred is part of an identified system.

In Figure 1B, the application provider sends an application load unit 111 to the interface device 105 and finally to IC card 103. The ALU includes the application itself and security data required to authenticate and protect the application code and associated data. The ALU is discussed specifically in Figure 2 and in connection with the other figures herein. The ALU 111 also preferably contains Application Load Certificate (ALC) 113 data which is sent from the Certification Authority (CA) 109 to the application provider 101. The Certification Authority manages the overall security of the system by providing an Application Load Certificate for each application which is to be loaded onto an IC card. The application provider 101 and the IC card 103 both have individual public/secret keys sets. The authentication and security processes will now be described.

Figure 2 shows a diagram illustrating the components of an Application Load Unit which is sent from the application loader to the IC card during the application load process. The Application Load Unit (ALU) 201 contains an Application Unit (AU) 203, an Application Unit Signature (AU_S) 205, a Key Transformation Unit (KTU) 207 and an Application Load Certificate (ALC) 209. The ALU 201 is formatted in a conventional format used during data transmission. AU 203 contains the application code and data which are to be stored on the IC card, some or all of which is encrypted to protect a secret portion or portions of the code and/or data. AU 203 is described in further detail in

connection with Figure 3.

AU_S 205 is the application code and data AU 203 digitally signed with the secret key of the application provider. The public key of the application provider is sent as part of the ALC 209 and is used to authenticate the application provider as the originator of the application. ALC 209 is made up of card identification information and the application provider's public key and is signed by the secret key of the certification authority. All these elements will be described in more detail below.

KTU 207 contains information relating to the encryption of the AU 203 (the code and data of the application) which allows the IC card to decrypt the designated portions so that the application and data can be accessed by the IC card but protects the data during transmission between the application provider and the IC card. KTU 207 is encrypted with the public key of the IC card for which the application is intended which ensures that only the intended IC card can decrypt the application code and data using the KTU information. This element will be described in connection with Figure 5.

Figure 3 shows a graphic representation of the Application Unit 203 which is part of the application load unit. The AU 203 contains both the program code and associated data which is to be loaded onto the IC card of the card user. The program code consists of a number of program instructions which will be executed by the microprocessor on the IC card. The program instructions can be written in any programming language which the operating system stored on the IC card can interpret.

For example, in the MULTOS system the program can be written in MEL™ (MULTOS Executable Language). Most applications have associated data which must be loaded onto the card. For instance, data which identifies the card user such as a person's name or account number may be loaded in a secure manner with the credit/debit application. An application provider may provide electronic cash represented by data as a promotion when installing an electronic purse application. Some or all of this data is desired to be kept secret from third parties. Additionally, the application code itself may be considered proprietary and portions may be desired to be kept secret from others. The use of a Key Transformation Unit (KTU) will allow an application provider to designate and encrypt selected portions of its application as confidential and protect it from third parties.

Application Unit portion 305 indicates the program code which is to be transferred from the application provider to the IC card. Application Unit portion 307 indicates the associated data which is to be transferred as part of the application to be loaded onto the IC card. In this example, three discrete areas of the application unit are shown to be encrypted using either single DES or triple DES. Any number of variations regarding the portions encrypted and the type of encryption can be employed using the techniques described herein.

In this example, encrypted location 309 shows the first portion of the Application Unit 203 which has been encrypted using a triple DES technique. The encryption process as described above involves using a symmetric key and the conventionally known DES-based algorithm to transform the data. The data can later be recovered by applying the key to a conventionally known DES-based

decryption algorithm. Encrypted location 311 shows a second portion of the application unit 203 which has been encrypted using triple DES. Encrypted location 313 shows a third portion which is encrypted using single DES. Single DES requires less computation to decrypt and takes up less space as part of the KTU as described below. If the application unit were intercepted by a third party while it was being transmitted from the application loader to the IC card, the encrypted portions could not be read unless the third party had the correct keys and decryption algorithm. That information, therefore, is protected in the KTU.

The KTU is used to allow the IC card for which the application and associated data is intended to decrypt the encrypted portions of the Application Unit by describing which portions of the application unit are encrypted, which encryption algorithm was used and the key or keys to be used to decipher the text. This information is highly confidential between the application provider and the intended IC card and therefore is protected in a manner unique to the intended card. In order to encrypt the KTU which is part of the overall ALU being transmitted, an individual key set for the particular intended IC card is used. The key set and its generation will now be described.

In accordance with an embodiment of the present invention, one of the security operations performed at the CA is to generate an individualized key set for each IC card which is stored on the card. The keys are used for off-card verification (i.e., to verify that the card is an authentic card) and for secure data transportation. The key generation process is shown generally in Figure 4. The key set is made up of three different key data items: the card's secret key which is

known only to the card, the card's public key which is stored on the card and the card's public key certificate which is the card's public key signed by the CA's secret key. The individual keys of the key set are described in more detail below.

Step 401 stores a card specific transport secret key for the individual IC card in the memory of the card. This secret key is generated by the CA from a standard asymmetric encryption technique such as RSA® and loaded onto the card via a card acceptance device. Once stored on the card, the CA deletes from its own memory any data relating to the secret key. Thus, only the card itself knows its secret key. The data element containing the secret key information in the card is called "mkd_sk" which stands for MULTOS key data secret key.

Step 403 stores a card specific transport public key for the individual IC card in the memory of the card. This public key is preferably generated by the CA from the asymmetric encryption technique used to produce the secret key in step 401. As with the secret key, once the public key is stored on the card, the CA (or other key provider) deletes from its systems the public key data so that the only copy of the public key is kept in the card. The data element containing the card's public key information is called "mkd_pk" which stands for MULTOS key data public key.

Step 405 stores a card specific transport public key certificate for the individual IC card in the memory of the card. The data element containing the card's public key certificate information is called "mkd_pk_c" which stands for MULTOS key data public key certificate. This public key certificate is preferably generated by signing the transport public key mkd_pk with the secret key of the

CA, indicated as follows:

$$mkd\_pk\_c = [mkd\_pk]_{CA\_sk}$$

which means the individual card's public key certificate is formed by applying the CA's secret key to the individual card's public key. The process is carried out at the CA. The public key certificate is retained by the CA so that it can regenerate the public key as needed.

A terminal can read the public key certificate from the IC cards to verify that the CA had signed and therefore approved the individual IC card. This is accomplished by verifying the public key certificate with the public component of the CA key set used to sign the mkd_pk.

Figure 5 is a graphic depiction of the contents of KTU 207, which contains Header portion 501 and KTU Ciphertext portion 503. As shown in Figure 5, header information 501 includes, for example, identifier or permissions information 505 such as the application_id_no (application identification number), mcd_no (IC card no) and/or msm_control_data_date (the date the IC card was issued). Additional identifiers could also be included. These identifiers allow the system to verify that the IC card which receives the ALU is the intended IC card. The permissions data is discussed in detail in the above referenced related application.

KTU Ciphertext 503 corresponds to KTU Plaintext (not encrypted) encrypted with the public key mkd_pk of the intended IC card as shown in box 507. The KTU Plaintext in further described in Figure 6. The public key mkd_pk is obtained from the intended IC card by the application provider. The public key

of an IC card is freely available to anyone and can be obtained directly from the card or from the CA. By encrypting the KTU Plaintext with the IC card public key, only the intended IC card can use its secret key of the public/secret key pair to decrypt the KTU Ciphertext. This means that only the intended IC card can determine the contents of the KTU plaint text, identify the encrypted portions of the application being loaded and use the keys to decrypt and recover the entire application and associate data. Because no other entity has the secret key of the IC card, the security and integrity of the program code and data being transmitted in ensured.

Figure 6 is a graphic representation of KTU Plaintext 601. KTU Plaintext 601 preferably includes identifier field 603, no_area_discriptors field 605, alg_id field 607, area_start field 609, area_length 611, key_length field 613, key_data field 615 and additional area and key fields depending upon the number of encrypted areas present in the Application Unit. Identifiers 603 contain identifying information of the Application Unit to which the KTU applies.

No_area_descriptors 605 indicates how many different portions of the AU have been encrypted. In the example of Figure 3, the number or area descriptors would be three. Field 607 contains the algorithm identifier for the first area which has been encrypted. The algorithm could be DES or triple DES, for example. Field 609 indicates the start of the first encrypted area. This indication could be an offset from the start of the AU. For example, the offset could by 100 which means that the first area starts at the 100th byte of the Application Unit. Field 611 indicates the area length for the first encrypted portions. This field allows the microprocessor on

the IC card to know how large an area has been encrypted and when coupled with the start of the area, allows the IC card microprocessor to decrypt the correct portion of the Application Unit. Filed 613 indicates the key length for the particular encrypted portion of the application unit. The length of the key will differ for different encryption techniques. The key length field allows the IC card to know the length of the key data. Field 615 indicates the key data for the particular encrypted portion. The key data is used with the algorithm identity and the location of the encoded portion to decode the encrypted portion. If more than one encrypted area is indicated, then additional data referring to the algorithm, start location, length, key length and key data will be present in the KTU Plaintext. While a number of fields have been described, not all the fields are necessary for the invention. The most important field, however, is the key data itself.

Figure 7 is a graphic representation of the Application Load Certificate (ALC) 209. ALC 209 includes a header 701 and the Application Provider Public Key 703. Header 701 and Application Provider Public Key 703 are then signed (encrypted) with the CA secret key. Thus, the ALC 209 must be provided by the CA to the application provider for each application loaded because only the CA knows the CA private key. Header 701 contains information regarding the application provider and the IC card for which the application is intended. The ALC 209 is placed in the correct ALU by the application provider which can use the identification information. Application Provider Public Key 703 is provided to the CA along with the identification data. The CA then signs this information after verifying its authenticity and returns the signed ALC to the application provider.

The IC card, when it receives the ALC 209 as part of the ALU 201, will verify the ALC 209 with the public key of the CA. This ensures that the CA signed the Application Load Certificate and that it is genuine. After verifying the information, the header identification information 701 is checked and the application provider public key is recovered. This public key will be used to verify that the application and code which is to be loaded onto the IC card originated with the proper application provider.

Figure 8 is a graphic representation of the use of the application provider's public key to verify the signature of the AU 205 in order to verify that AU 203 was signed by the application provider. AU signature 205 is verified with the Application Provider Public Key 801 and compared with AU 203. If the data blocks match, then the IC card has verified that the application provider signed (encrypted) the application unit and the application is genuine. This authentication is valid because only the application provider has its own secret key. The IC card can process this information efficiently because the application provider's public key is provided to it as part of the Application Load Certificate 209 which is signed by the CA. Therefore, it does not need to retrieve the public key from an external location to authenticate the application.

Figure 9 shows a flow chart of the steps for processing the Application Load Unit when it is received by the IC card. Prior to receiving the ALU, identity checks as to the identity of the IC card can be performed if desired. The ALU processing techniques provide a number of further verifications including verifying that the application being loaded is: (1) from the correct application

provider, (2) being loaded on the intended card and (3) certified by the CA. The ALU processing techniques also allow the transportation of transport decryption keys which enable the IC card to decrypt portions of the program code and associated data in a secure manner. In step 901, the IC card receives the ALU from the application provider. The ALU can be transmitted via a terminal connection, contactless connection, telephone, computer, intranet, Internet or any other communication means. The ALU is placed in an I/O buffer of the IC card along with header information indicating the starting addresses of AU 203, AU signed 205, the KTU 207 and ALC 209. Alternatively, the IC card could determine the relative address locations of these four units.

Step 903 verifies the ALC 209 with the CA public key. Each IC card preferably stores in its memory a copy of the CA public key because it is used in many transactions. Alternatively, the IC card could obtain the public key from a known storage location. If the CA public key verifies the ALC 209 properly, then the IC card has verified that the CA has signed the ALC 209 with its secret key and thus the Application Load Certificate is proper. If the IC card cannot verify the ALC properly, then the ALC was not signed by the CA and the certificate is not proper. The application loading process would then end.

Step 905 then checks the identity of IC card against the identification information sent in the Application Load Certificate to make sure the card is intended to receive the application. This permissions checking is described in the related patent application identified above. If there is no match of identification data, the application loading process ends. If the identification data does match,

then the process continues.

Step 907 uses the application providers public key which was recovered from the verified ALC to verify AU signature 205. When the ALU was generated by the application provider, the application unit 203 was signed with the application provider's secret key to authenticate that the application was provided by the correct application provider. The application provider then provides its public key to IC card through the ALC. The IC card then verifies the AU signature 205. If the two data blocks match, then the ALU is verified as being generated by the application provider. Because the application provider's public key is part of the ALC which is signed by the CA, the CA can make sure that the proper public key has been provided to the IC card. This unique key interaction between the application provider, CA and the intended IC card ensures that no counterfeit or unapproved applications or data are loaded onto an IC card which is part of the secure system.

Step 911 then processes a KTU authentication check which further verifies that only the intended card has received the application. The KTU authentication check makes sure that if a third party does somehow intercept the ALU, the third party cannot read the enciphered portions of the AU and cannot retrieve the keys to decrypt the AU. This step is further explained in Figure 10.

Figure 10 shows the steps of the KTU Authentication process. Step 1001, which is shown in dashed lines because it is preferably optional, checks the identification of the IC card a second time. The identification information can be sent as part of the KTU data. However, this check is optional as it has already

been performed once in step 905.

Step 1003 then decrypts KTU ciphertext 503 using the IC card's secret key (mkd_sk). The KTU Plaintext was previously encrypted using the intended card's public key (mkd_pk). This means that only the holder of the intended card's secret key could decrypt the encrypted message. The application provider obtains the intended IC card's public key either from the IC card itself (See Figure 4 and related text for a discussion of the mkd key set) or from a database holding the public keys. If the IC card cannot decrypt the KTU ciphertext properly then the KTU is not meant for that card and the application loading process halts. If the IC card does properly decipher the KTU ciphertext, then the process continues.

Step 1005 identifies an encrypted area of the application unit (AU). In the example of the KTU Plaintext described in connection with Figure 6, the IC card uses a relative starting address and area length field to determine the encrypted portion. Step 1005 also identifies which encryption technique was used to encrypt the identified portion so that the proper decryption technique can be used. For example, the technique could by single or triple DES. Alternatively, the technique could be a default technique used in the system and need not be identified.

Step 1007 then retrieves the key from KTU Plaintext and decrypts the identified portion with the identified decryption technique. This allows the IC card to have the decrypted portion of the AU which it will store in its EEPROM once all the encrypted portions have been decrypted.

Step 1009 checks if there are any other additional encrypted areas.

In the example described in Figure 3, there are three encrypted areas. The number of encrypted areas was a field in the example of Figure 6. However, the number of portions can be determined using other conventional means. If there are additional encrypted portions, the process jumps to step 1005. If there are no additional encrypted portions, then the process continues with step 1011.

Step 1011 then loads the decrypted AU into the memory of the IC card. The ALU has passed all of the authentication and decryption checks and the application can now properly reside on the IC card and be executed and used by the card user. While the different checks have been presented in a particular order in Figures 9 and 10, the checks can be performed in any order. While all of the described techniques used in conjunction with the ALU provide the best security, one or more of the individual techniques could be used for their individual purposes or combined with other conventional security techniques.

Figure 11 shows an example of a block diagram of an IC card chip upon which an ALU can be loaded and processed. An integrated circuit is located on an IC card for use. The IC card preferably includes a central processing unit 1101, a RAM 1103, an EEPROM 1105, a ROM 1107, a timer 1109, control logic 1111, an I/O port 1113 and security circuitry 1115, which are connected together by a conventional data bus.

Control logic 1111 in memory cards provides sufficient sequencing and switching to handle read-write access to the card's memory through the input/output ports. CPU 1101 with its control logic can perform calculations, access memory locations, modify memory contents, and manage input/output ports.

Some cards have a coprocessor for handling complex computations like cryptographic operations. Input/output ports 1113 are used under the control of a CPU and control logic, for communications between the card and a card interface device. Timer 1109 (which generates or provides a clock pulse) drives the control logic 1111 and CPU 1101 through the sequence of steps that accomplish memory access, memory reading or writing, processing, and data communication. A timer may be used to provide application features such as call duration. Security circuitry 1115 includes fusible links that connect the input/output lines to internal circuitry as required for testing during manufacture, but which are destroyed ("blown") upon completion of testing to prevent later access. The AU data after the ALU has been authenticated and verified is stored in EEPROM 1105. The IC card private key will be stored in a secure memory location. The IC card public key and public key certificate is preferably stored in EEPROM 1105. The authentication process as described herein is performed by the CPU 1101.

Figure 11 also shows a possible configuration for the application provider, transmitting entity and for the CA. CPU 1101 present in the application provider encrypts the necessary information using encryption techniques described herein and performs the necessary data operations. CPU 1101 present in the certification authority is used to sign the Application Load Certificate and the public key certificate as described herein.

The foregoing merely illustrates the principles of the invention. It will thus be appreciated that those skilled in the art will be able to devise numerous systems and methods which, although not explicitly shown or described herein,

embody the principles of the invention and are thus within the spirit and scope of the invention.

For example, while loading an application is discussed herein, the same secure loading processes can apply to transmitting other types of data such as data blocks, database files, word processing documents or any other type of data need to be transmitted in a secure manner.

The scope of the present disclosure includes any novel feature or combination of features disclosed therein either explicitly or implicitly or any generalisation thereof irrespective of whether or not it relates to the claimed invention or mitigates any or all of the problems addressed by the present invention. The application hereby gives notice that new claims may be formulated to such features during the prosecution of this application or of any such further application derived therefrom. In particular, with reference to the appended claims, features from dependant claims may be combined with those of the independent claims in any appropriate manner and not merely in the specific combinations enumerated in the claims.

ANNEX A TO THE DESCRIPTION

ANNEX A

KEY TRANSFORMATION UNIT FOR AN IC CARD

ANNEX A TO THE DESCRIPTION

BACKGROUND OF INVENTION

Integrated circuit ("IC") cards are becoming increasingly used for many different purposes in the world today. An IC card (also called a smart card) typically is the size of a conventional credit card which contains a computer chip including a microprocessor, read-only-memory (ROM), electrically erasable programmable read-only-memory (EEPROM), an Input/Output (I/O) mechanism and other circuitry to support the microprocessor in its operations. An IC card may contain a single application or may contain multiple independent applications in its memory. MULTOS™ is a multiple application operating system which runs on IC cards. among other platforms, and allows multiple applications to be executed on the card itself. This allows a card user to run many programs stored in the card (for example, credit/debit, electronic money/purse and/or loyalty applications) irrespective of the type of terminal (i.e., ATM, telephone and/or POS) in which the card is inserted for use.

A conventional single application IC card, such as a telephone card or an electronic cash card, is loaded with a single application when it is manufactured and before it is given to a card user. That application, however, cannot be modified or changed after the card is issued even if the modification is desired by the card user or card issuer. Moreover, if a card user wanted a variety of application functions to be performed by IC cards issued to him or her, such as both an electronic purse and a credit/debit function, the card user would be required to carry multiple physical cards on his or her person, which would be quite cumbersome and inconvenient. If an application developer or card user desired two

different applications to interact or exchange data with each other, such as a purse application interacting with a frequent flyer loyalty application, the card user would be forced to swap multiple cards in and out of the card-receiving terminal, making the transaction difficult, lengthy and inconvenient.

Therefore, it is beneficial to store multiple applications on the same IC card. For example, a card user may have both a purse application and a credit/debit application on the same card so that the user could select which type of payment (by electronic cash or credit card) to use to make a purchase. Multiple applications could be provided to an IC card if sufficient memory exists and an operating system capable of supporting multiple applications is present on the card. Although multiple applications could be preselected and placed in the memory of the card during its production stage, it would also be beneficial to have the ability to load and delete applications for the card post-production as needed.

The increased flexibility and power of storing multiple applications on a single card create new challenges to be overcome concerning the integrity and security of the information (including application code and associated data) exchanged between the individual card and the application provider as well as within the entire system when loading and deleting applications. It would be beneficial to have the capability in the IC card system to exchange data among cards, card issuers, system operators and application providers securely and to load and delete applications securely at any time from a local terminal or remotely over a telephone line, Internet or intranet connection or other data conduit. Because these data transmission lines are not typically secure lines, a number of security and

ANNEX A TO THE DESCRIPTION

entity authentication techniques must be implemented to make sure that applications being sent over the transmission lines are not tampered with and are only loaded on the intended cards.

As mentioned, it is important -- particularly where there is a continuing wide availability of new applications to the cardholder -- that the system has the capability of adding applications onto the IC card subsequent to issuance. This is necessary to protect the longevity of the IC cards; otherwise, once an application becomes outdated, the card would be useless. It would be beneficial to allow the addition of applications from a remote location as well as from a direct connection to an application provider's terminal. For example, it would be beneficial for a card user to be able to plug his IC card into his home computer and download an application over the Internet. This type of remote loading of applications raises a number of security risks when transmitting the application code and related data over an unsecured communications line such as the Internet. At least three issues need to be addressed in a system which provides such a capability.

The first issue is to make sure that the IC card receiving the application is the intended IC card and not another IC card. The second issue is determining how the IC card can authenticate that the application came from the proper application provider and not an unknown third party. The third issue concerns preventing third parties from reading the application and making an unauthorized copy. If a portion of the application is encrypted to address the latter issue, the intended IC card needs to have access to the correct key to decrypt the application. In a system with many IC cards and additionally many application

providers, a secure key transfer technique is required so that the intended IC card can use the correct key for the application which is received. These concerns are raised by both remote application loading as well as local terminal application loading.

Accordingly, it is an object of this invention to provide a key transfer and authentication technique and specifically to provide a secure IC-card system that allows for the secure transfer of smart card applications which may be loaded onto IC cards.

SUMMARY OF THE INVENTION

These and other objectives are achieved by the present invention which provides an IC card system and method for securely loading an application onto an IC card including providing a secret and public key pair for the IC card, encrypting at least a portion of the application using a transfer key, encrypting the transfer key using the IC card's public key to form a key transformation unit, transmitting the encrypted application and the key transformation unit to the IC card, decrypting the key transformation unit using the IC card's secret key to provide the transfer key, decrypting the encrypted application using the provided transfer key and storing the decrypted application on the IC card.

In a preferred embodiment, the secure loading system and method allows the application provider to encrypt two or more portions of the application to be transmitted with two or more different keys, encrypt the two or more keys with the public key of the IC card to form a key transformation unit including the

ANNEX A TO THE DESCRIPTION

locations of the encrypted portions. Both the encrypted application and the key transformation unit are sent to the IC card. Because the decryption keys are encrypted with the IC card's public key, only the IC card's secret key can decrypt the key transformation unit. The transfer keys and the locations of the encrypted portions are recovered from the decrypted key transformation unit and the application is decrypted using the recovered transfer keys. This ensures that only the intended IC card can decrypt and use the application which was transmitted to that IC card.

In a preferred embodiment, an application load certificate is also sent to the IC card which is receiving the application. The application load certificate contains the public key of the application provider encrypted by the secret key of the certificate authority ("CA"), or the entity that manages the overall security of the IC card system. The IC card then uses a certificate authority public key to make sure that the certificate was valid by attempting to verify the application load certificate with the CA's public key. The IC card then uses the recovered application provider's public key to verify that the application provider was in fact the originator of the application by verifying the sent application signature generated with the application provider's corresponding secret key.

BRIEF DESCRIPTION OF THE DRAWINGS

Further objects, features and advantages of the invention will become apparent from the following detailed description taken in conjunction with the accompanying figures showing illustrative embodiments of the invention, in which

ANNEX A TO THE DESCRIPTION

Fig. 1 is block diagram of the application loading system which loads an application from an application provider to an IC card;

Fig. 2 is a graphic representation of the contents of an Application Loading Unit;

Fig. 3 is a graphic representation of an Application Unit;

Fig. 4 is a flow chart of the steps for providing an individual key set for an IC card;

Fig. 5 is a graphic representation of a Key Transformation Unit;

Fig. 6 is a graphic representation of a Key Transformation Unit plaintext;

Fig. 7 is a graphic representation of the Application Load Certificate;

Fig. 8 is a graphic representation of the Application Unit being decrypted;

Fig. 9 is a flowchart illustrating the steps undertaken in processing the Application Load Unit;

Fig. 10 is a flowchart illustrating the steps undertaken in processing the KTU; and

Fig. 11 is a block diagram showing the components of an IC card which can receive and process and Application Load Unit.

Throughout the figures, the same reference numerals and characters, unless otherwise stated, are used to denote like features, elements, components or portions of the illustrated embodiments. Moreover, while the subject invention will now be described in detail with reference to the figures, it is done so in connection

ANNEX A TO THE DESCRIPTION

with the illustrative embodiments. It is intended that changes and modifications can be made to the described embodiments without departing from the true scope and spirit of the subject invention as defined by the appended claims.

DETAILED DESCRIPTION OF THE INVENTION

It is beneficial to have the capability to load applications onto IC cards containing multiple application operating systems at any time during the lifetime of the IC card. This flexibility allows a user of a card to periodically add new applications to the IC card and also allows older applications to be updated with newer versions of the application when they are released. For example, a card user may start with an IC card that contains a purse, or electronic cash application (e.g., MONDEX™), being stored on his IC card. Some time after the user has the card, he or she may load an additional application onto the card such as a credit/debit application. Some time after loading the credit/debit application on the card, a new version of the credit/debit application may become available and the card user should be able to erase the old application on his IC card and replace it with the new version of the credit/debit application which may contain additional features.

The flexibility of loading applications at different times during the IC card's life cycle creates security issues with the process of loading applications onto the card. In a multiple application operating system environment, it is beneficial to be able to load applications both at terminals, such as a bank ATM machine, as well as over remote communication links, such as telephone lines, cable

lines, the Internet, satellite or other communications means. When loading applications onto an IC card, the application provider and the card issuer (which could be the same entity) needs to provide security regarding the applications to be loaded. First, the application provider must make sure the application is only sent to the correct card user who is intended to receive the application. One solution to this problem is addressed in a related application entitled "Secure Multi-Application IC Card System Having Selective Loading and Deleting Capability" by Everett et al., filed February 12, 1998 and assigned to Mondex International, which is hereby incorporated by reference. Two additional security concerns also need to be addressed when loading an application from a remote source, or even from a local terminal, onto an IC card. First, the source of the application must be authenticated as the proper originator so that applications which may contain viruses or simply take up the limited storage memory in an IC card are not allowed to be loaded onto an IC card. Second, the application and associated data may contain private or trade secret information which needs to be encrypted so other people cannot view the contents of the encrypted application code and data. A portion of the application code and data may be secret while other portions are not. These concerns of authentication and protecting the contents of some or all of the application and associated data being loaded onto a card is addressed herein.

A number of encryption/decryption techniques are described herein. There are two basic types of encryption, symmetric encryption and asymmetric encryption. Symmetric encryption uses a secret key as part of a mathematical formula which encrypts data by transforming the data using the formula and key.

After the data is encrypted, another party can decrypt the encrypted data using the same secret key with a related decryption algorithm. Thus the same key is used for encryption and decryption so the technique is symmetric. A conventional example of a symmetric algorithm is DES.

Asymmetric encryption techniques use two different keys of a pair for encrypting and decrypting information. The two keys are normally referred to as a private or secret key and a public key. When data is encrypted with one key of the pair, the other key is used to decrypt the data. If a sender of data signs the data with his secret key, anyone with the public key can verify the message. Since public keys are typically known to the public, the contents of a data signed with a secret key cannot be protected but the origination of the data can be verified by determining if a particular secret key signed the data. This authentication process is termed a digital signature. If person A wanted to authenticate a message he was sending to person B, the person A would sign the document with his secret key. When person B received the message, he would use person A's public key to decipher the message. If the message was readable after the public key was applied to it, person B would know that the document was signed with secret key of person A. Thus, the origin of the message has been authenticated.

The asymmetric key set can also be used to protect the contents of a message. If person A wanted to send an encrypted message to person B that no one else could read, he would encrypt the data or message with person B's public key and send it to person B. Now only the holder of B's secret key could decrypt the data. If a combination of keys is used, a person could both authenticate and

ANNEX A TO THE DESCRIPTION

encrypt the message. The asymmetric pair of keys has some powerful applications with respect to card security and is more robust than symmetric encryption. However, asymmetric encryption is more processor costly that symmetric encryption. A example of an asymmetric encryption method is RSA.

A hybrid of symmetric encryption which makes the encryption method more powerful is to encrypt data using two symmetric keys. This technique is called triple DES which encodes data with symmetric key 1, decodes the data using symmetric key 2 (which in effect further encodes the data) and then further encodes the data using key 1 again. Once the data has arrived at its destination, key 1 is used to decode the data, key 2 is used to encode the data, and key 1 is used to decode the data. These extra steps of encoding and decoding make the technique more powerful and more difficult to properly decipher without both keys.

Figure 1 shows a block diagram of the entities used in a secure remote application loading process. The application provider 101 can be a card issuer, bank or other entity which provides application loading services. The application provider 101 initiates an application loading process onto IC card 103. Application Provider 101 is connected to data conduit 107 which is connected to interface device 105 (e.g., a terminal that communicates with an IC card). Data conduit 107 can be a telephone line, an intranet, the Internet, a satellite link or any other type of communications link. The application provider 101, which is remotely located from the IC card 103, desires to send and load an application to the IC card. However, because the data link is an open link and subject to third parties possibly intercepting or replacing applications being transmitted, security

ANNEX A TO THE DESCRIPTION

measures which authenticate the application itself, the application provider and the IC card must be used to ensure the integrity of the system. The Certificate Authority 109 may also be used to help authenticate that some data being transferred is part of an identified system.

In Figure 1, the application provider sends an application load unit 111 to the interface device 105 and finally to IC card 103. The ALU includes the application itself and security data required to authenticate and protect the application code and associated data. The ALU is discussed specifically in Figure 2 and in connection with the other figures herein. The ALU 111 also preferably contains Application Load Certificate (ALC) 113 data which is sent from the Certification Authority (CA) 109 to the application provider 101. The Certification Authority manages the overall security of the system by providing an Application Load Certificate for each application which is to be loaded onto an IC card. The application provider 101 and the IC card 103 both have individual public/secret keys sets provided to them. The authentication and security processes will now be described.

Figure 2 shows a diagram illustrating the components of an Application Load Unit which is sent from the application loader to the IC card during the application load process. The Application Load Unit (ALU) 201 contains an Application Unit (AU) 203. an Application Unit Signature (AU_S) 205, a Key Transformation Unit (KTU) 207 and an Application Load Certificate (ALC) 209. The ALU 201 is formatted in a conventional format used during data transmission. AU 203 contains the application code and data which are to be stored

ANNEX A TO THE DESCRIPTION

on the IC card, some or all of which is encrypted to protect a secret portion or portions of the code and/or data. AU 203 is described in further detail in connection with Figure 3.

AU_S 205 is the application code and data AU 203 digitally signed with the secret key of the application provider. The public key of the application provider is sent as part of the ALC 209 and is used to authenticate the application provider as the originator of the application. ALC 209 is made up of card identification information and the application provider's public key and is signed by the secret key of the certification authority. All these elements will be described in more detail below.

KTU 207 contains information relating to the encryption of the AU 203 (the code and data of the application) which allows the IC card to decrypt the designated portions so that the application and data can be accessed by the IC card but protects the data during transmission between the application provider and the IC card. KTU 207 is signed with a public key of the IC card for which the application is intended which ensures that only the intended IC card can decrypt the application code and data using the KTU information. This element will be described in connection with Figure 5.

Figure 3 shows a graphic representation of the Application Unit 203 which is part of the application load unit. The AU 203 contains both the program code and associated data which is to be loaded onto the IC card of the card user. The program code consists of a number of program instructions which will be executed by the microprocessor on the IC card. The program instructions can be

ANNEX A TO THE DESCRIPTION

written in any programming language which the operating system stored on the IC card can interpret.

For example, in the MULTOS system the program can be written in MEL™ (MULTOS Executable Language). Most applications have associated data which must be loaded onto the card. For instance, data which identifies the card user such as a person's name or account number may be loaded in a secure manner with the credit/debit application. An application provider may provide electronic cash represented by data as a promotion when installing an electronic purse application. Some or all of this data is desired to be kept secret from third parties. Additionally, the application code itself may be considered proprietary and portions may be desired to be kept secret from others. The use of a Key Transformation Unit (KTU) will allow an application provider to designate and encrypt selected portions of its application as confidential and protect it from third parties.

Application Unit portion 305 indicates the program code which is to be transferred from the application provider to the IC card. Application Unit portion 307 indicates the associated data which is to be transferred as part of the application to be loaded onto the IC card. In this example, three discrete areas of the application unit are shown to be encrypted using either single DES or triple DES. Any number of variations regarding the portions encrypted and the type of encryption can be employed using the techniques described herein.

In this example, encrypted location 309 shows the first portion of the Application Unit 203 which has been encrypted using a triple DES technique. The encryption process as described above involves using a symmetrical key and the

ANNEX A TO THE DESCRIPTION

conventionally known DES algorithm to transform the data. The data can later be recovered by applying the key to the known DES algorithm. Encrypted location 311 shows a second portion of the application unit 203 which has been encrypted using triple DES. Encrypted location 313 shows a third portion which is encrypted using single DES. Single DES requires less computation to decrypt and takes up less space as part of the KTU as described below. If the application unit were intercepted by a third party while it was being transmitted from the application loader to the IC card, the encrypted portions could not be read unless the third party had the correct keys. That information, therefore, is protected in the KTU.

The KTU is used to allow the IC card for which the application and associated data is intended to decrypt the encrypted portions of the Application Unit by describing which portions of the application unit are encrypted, which encryption algorithm was used and the key or keys to be used to decipher the text. This information is highly confidential between the application provider and the intended IC card and therefore is protected in a manner unique to the intended card. In order to encrypt the KTU which is part of the overall ALU being transmitted, an individual key set for the particular intended IC card is used. The key set and its generation will now be described.

One of the security operations performed at the CA is to generate an individualized key set for each IC card which is stored on the card. The keys are used for off-card verification (i.e., to verify that the card is an authentic card) and for secure data transportation. The key generation process is shown generally in Figure 4. The key set is made up of three different key data items: the card's

ANNEX A TO THE DESCRIPTION

secret key which is known only to the card. the card's public key which is stored on the card and the card's public key certificate which is the card's public key signed by one of the CA's secret keys. The individual keys of the key set are described in more detail below.

Step 401 stores a card specific transport secret key for the individual IC card in the memory of the card. This secret key is generated by the CA and loaded onto the card via a card acceptance device. Once stored on the card, the CA deletes from its own memory any data relating to the secret key. Thus, only the card itself knows its secret key. The data element containing the secret key information in the card is called "mkd_sk" which stands for MULTOS key data secret key.

Step 403 stores a card specific transport public key for the individual IC card in the memory of the card. This public key is preferably generated by the CA from the asymmetric encryption technique used to produce the secret key in step 401. The data element containing the card's public key information is called "mkd_pk" which stands for MULTOS key data public key.

Step 405 stores a card specific transport public key certificate for the individual IC card in the memory of the card. The data element containing the card's public key certificate information is called "mkd_pk_c" which stands for MULTOS key data public key certificate. This public key certificate is preferably generated by encrypting the transport public key mkd_pk with the secret key of the CA, indicated as follows:

$$mkd\_pk\_c = [mkd\_pk]_{CA\_sk}$$

which means the individual card's public key certificate is formed by applying the CA's secret key to the individual card's public key. The process is carried out at the CA. The public key certificate is retained by the CA so that it can regenerate the public key as needed.

A terminal can read the public key certificate from the IC cards to verify that the CA had signed and therefore approved the individual IC card. This is accomplished by verifying the public key certificate with the public component of the CA key set used to sign the mkd_pk. The decrypted public key certificate can then be compared with the public key to verify that the key certificate was certified (signed) by the CA.

Figure 5 is a graphic depiction of the contents of KTU 207, which contains Header portion 501 and KTU Ciphertext portion 503. As shown in Figure 5, header information 501 includes, for example, identifier or permissions information 505 such as the application_id_no (application identification number), mcd_no (IC card no) and/or msm_control_data_date (the date the IC card was issued). Additional identifiers could also be included. These identifiers allow the system to verify that the IC card which receives the ALU is the intended IC card. The permissions data is discussed in detail in the above referenced related application.

KTU Ciphertext 503 corresponds to KTU Plaintext (not encrypted) encrypted with the public key mkd_pk of the intended IC card as shown in box 507. The KTU Plaintext in further described in Figure 6. The public key mkd_pk is obtained from the intended IC card by the application provider. The public key

ANNEX A TO THE DESCRIPTION

of an IC card is freely available to anyone and can be obtained directly from the card or from the CA. By signing the KTU Plaintext with the IC card public key, only the intended IC card can use its secret key of the public/secret key pair to decrypt the KTU Ciphertext. This means that only the intended IC card can determine the contents of the KTU plaint text, identify the encrypted portions of the application being loaded and use the keys provided to decrypt and recover the entire application and associate data. Because no other entity has the secret key of the IC card, the security and integrity of the program code and data being transmitted in ensured.

Figure 6 is a graphic representation of KTU Plaintext 601. KTU Plaintext 601 preferably includes identifier field 603, no_area_discriptors field 605, alg_id field 607, area_start field 609, area_length 611, key_length field 613, key_data field 615 and additional area and key fields depending upon the number of encrypted areas present in the Application Unit. Identifiers 603 contain identifying information of the Application Unit to which the KTU applies.

No_area_descriptors 605 indicates how many different portions of the AU have been encrypted. In the example of Figure 3, the number or area descriptors would be three. Field 607 contains the algorithm identifier for the first area which has been encrypted. The algorithm could be DES or triple DES, for example. Field 609 indicates the start of the first encrypted area. This indication could be an offset from the start of the AU. For example, the offset could be 100 which means that the first area starts at the 100th byte of the Application Unit. Field 611 indicates the area length for the first encrypted portions. This field allows the microprocessor on

ANNEX A TO THE DESCRIPTION

the IC card to know how large an area has been encrypted and when coupled with the start of the area, allows the IC card microprocessor to decrypt the correct portion of the Application Unit. Filed 613 indicates the key length for the particular encrypted portion of the application unit. The length of the key will differ for different encryption techniques. The key length field allows the IC card to know the length of the key data. Field 615 indicates the key data for the particular encrypted portion. The key data is used with the algorithm identity and the location of the encoded portion to decode the encrypted portion. If more than one encrypted area is indicated, then additional data referring of the algorithm, start location, length, key length and key data will be present in the KTU Plaintext. While a number of fields have been described, not all the fields are necessary for the invention. The most important field, however, is the key data itself.

Figure 7 is a graphic representation of the Application Load Certificate (ALC) 209. ALC 209 includes a header 701 and the Application Provider Public Key 703. Header 701 and Application Provider Public Key 703 are then signed (encrypted) with the CA secret key. Thus, the ALC 209 must be provided by the CA to the application provider for each application loaded because only the CA knows the CA private key. Header 701 contains information regarding the application provider and the IC card for which the application is intended. The ALC 209 is placed in the correct ALU by the application provider which can use the identification information. Application Provider Public Key 703 is provided to the CA along with the identification data. The CA then signs this information after verifying its authenticity and returns the signed ALC to the application provider.

The IC card, when it receives the ALC 209 as part of the ALU 201, will open the ALC 209 with the public key of the CA. This ensures that the CA signed the application load certificate and that it is genuine. After decrypting the information, the header identification information 701 is checked and the application provider public key is recovered. This public key will be used to verify that the application and code which is to be loaded onto the IC card originated with the proper application provider.

Figure 8 is a graphic representation of the use of the application provider's public key to decrypt the signed AU 205 in order to verify that AU 203 was signed by the application provider. AU signed 205 is verified with the Application Provider Public Key 801. The recovered AU 803 is then compared with AU 203. If the data blocks match, then the IC card has verified that the application provider signed (encrypted) the application unit and the application is genuine. This authentication is valid because only the application provider has its own secret key. The IC card can process this information because the application provider's public key is provided to it as part of the application load certificate 209 which is signed by the CA. Therefore, it does not need to retrieve the public key from an external location to authenticate the application.

Figure 9 shows a flow chart of the steps for processing the Application Load Unit when it is received by the IC card. Prior to receiving the ALU, identity checks as to the identity of the IC card can be performed if desired. The ALU processing techniques provide a number of further verifications including verifying that the application being loaded is: (1) from the correct application

ANNEX A TO THE DESCRIPTION

provider, (2) being loaded on the intended card and (3) certified by the CA. The ALU processing techniques also allow the transportation of transport decryption keys which enable the IC card to decrypt portions of the program code and associated data in a secure manner. In step 901, the IC card receives the ALU from the application provider. The ALU can be transmitted via a terminal connection, contactless connection, telephone, computer, intranet, Internet or any other communication means. The ALU is placed in the EEPROM of the IC card along with header information indicating the starting addresses of AU 203, AU signed 205, the KTU 207 and ALC 209. Alternatively, the IC card could determine the relative address locations of these four units.

Step 903 decrypts the ALC 209 with the CA public key. Each IC card preferably stores in its memory a copy of the CA public key because it is used in many transactions. Alternatively, the IC card could obtain the public key from a known storage location. If the CA public key successfully verifies the ALC 209, then the IC card has verified that the CA has signed the ALC 209 with its secret key and thus the Application Load Certificate is proper. If the IC card cannot verify the ALC successfully, then the ALC was not signed by the CA and the certificate is not proper. The application loading process would then end.

Step 905 then checks the identity of IC card against the identification information sent in the application load certificate to make sure the card is intended to receive the application. This permissions checking is described in the related patent application identified above. If there is no match of identification data, the application loading process ends. If the identification data does match, then the

ANNEX A TO THE DESCRIPTION

process continues.

Step 907 uses the application providers public key which was recovered from the verified ALC to verify the AU signature 205. When the ALU was generated by the application provider, the application unit 203 was signed with the application provider's secret key. The application provider then provides its public key to IC card through the ALC. The IC card then verifies the AU signed 205. If the ALU is successfully verified, then it is accepted as having been generated by the application provider. Because the application provider's public key is part of the ALC which is signed by the CA. the CA can make sure that the proper public key has been provided to the IC card. This unique key interaction between the application provider, CA and the intended IC card ensures that no counterfeit or unapproved applications or data are loaded onto an IC card which is part of the secure system.

Step 911 then processes a KTU authentication check which further verifies that only the intended card has received the application. The KTU authentication check makes sure that if a third party does somehow intercept the ALU, the third party cannot read the enciphered portions of the AU and cannot retrieve the keys to decrypt the AU. This step is further explained in Figure 10.

Figure 10 shows the steps of the KTU Authentication process. Step 1001, which is shown in dashed lines because it is preferably optional, checks the identification of the IC card a second time. The identification information can be sent as part of the KTU data. However, this check is optional as it has already been performed once in step 905.

ANNEX A TO THE DESCRIPTION

Step 1003 then decrypts KTU ciphertext 503 using the IC card's secret key (mkd_sk). The KTU Plaintext was previously encrypted using the intended card's public key (mkd_pk). This means that only the holder of the intended card's secret key could decrypt the encrypted message. The application provider obtains the intended IC card's public key either from the IC card itself (See Figure 4 and related text for a discussion of the mkd key set) or from a database holding the public keys. If the IC card cannot decrypt the KTU ciphertext properly then the KTU is not meant for that card and the application loading process halts. If the IC card does properly decipher the KTU ciphertext, then the process continues.

Step 1005 identifies an encrypted area of the application unit (AU). In the example of the KTU Plaintext described in connection with Figure 6, the IC card uses a relative starting address and area length field to determine the encrypted portion. Step 1005 also identifies which encryption technique was used to encrypt the identified portion so that the proper decryption technique can be used. For example, the technique could by single or triple DES. Alternatively, the technique could be a default technique used in the system and need not be identified.

Step 1007 then retrieves the key from KTU Plaintext and decrypts the identified portion with the identified decryption technique. This allows the IC card to have the decrypted portion of the AU which it will store in its static memory once all the encrypted portions have been decrypted.

Step 1009 checks if there are any other additional encrypted areas. In the example described in Figure 3, there are three encrypted areas. The number

of encrypted areas was a field in the example of Figure 6. However, the number of portions can be determined using other conventional means. If there are additional encrypted portions, the process jumps to step 1005. If there are no additional encrypted portions, then the process continues with step 1011.

Step 1011 then loads the decrypted AU into the memory of the IC card. The ALU has passed all of the authentication and decryption checks and the application can now properly reside on the IC card and be executed and used by the card user. While the different checks have been presented in a particular order in Figures 9 and 10, the checks can be performed in any order. While all of the described techniques used in conjunction with the ALU provide the best security, one or more of the individual techniques could be used for their individual purposes or combined with other conventional security techniques.

Figure 11 shows an example of a block diagram of an IC card chip upon which an ALU can be loaded and processed. An integrated circuit is located on an IC card for use. The IC card preferably includes a central processing unit 1101, a RAM 1103, an EEPROM 1105, a ROM 1107, a timer 1109, control logic unit 1111, an I/O port 1113 and security circuitry 1115, which are connected together by a conventional data bus.

Control logic 1111 in memory cards provides sufficient sequencing and switching to handle read-write access to the card's memory through the input/output ports. CPU 1101 with its control logic can perform calculations, access memory locations, modify memory contents, and manage input/output ports. Some cards have a coprocessor for handling complex computations like performing

cryptographic operations. Input/output ports 1113 are used under the control of a CPU and control logic, for communications between the card and a card interface device. Timer 1109 (which generates or provides a clock pulse) drives the control logic 1111 and CPU 1101 through the sequence of steps that accomplish memory access, memory reading or writing, processing, and data communication. A timer may be used to provide application features such as call duration. Security circuitry 1115 includes fusible links that connect the input/output lines to internal circuitry as required for testing during manufacture, but which are destroyed ("blown") upon completion of testing to prevent later access. The AU data after the ALU has been authenticated and verified is stored in EEPROM 1105. The authentication process as described herein is performed by the CPU 1101.

Figure 11 also shows a possible configuration for the integrated circuit chip for the application provider and for the certification authority. CPU 1101 present in the IC chip for the application provider encrypts the necessary information using encryption techniques described herein and performs the necessary data operations. CPU 1101 at the certification authority is used to sign the Application Load Certificate as described herein.

The foregoing merely illustrates the principles of the invention. It will thus be appreciated that those skilled in the art will be able to devise numerous systems and methods which, although not explicitly shown or described herein, embody the principles of the invention and are thus within the spirit and scope of the invention.

For example, while loading an application is discussed herein, the

ANNEX A TO THE DESCRIPTION

same secure loading process can apply to transmitting other types of data such as data blocks, database files, word processing documents or any other type of data need to be transmitted in a secure manner.

I CLAIM:

ANNEX A TO THE DESCRIPTION

1. A method for securely loading an application onto an IC card comprising the steps of:

providing a secret key and public key pair for said IC card;

encrypting at least a portion of said application using a transfer key;

encrypting said transfer key using said IC card's public key to form a key transformation unit;

transmitting said encrypted application and said key transformation unit to said IC card;

decrypting said key transformation unit using said IC card's secret key to recover said transfer key; and

decrypting said encrypted application using said recovered transfer key.

2. The method of claim 1, further including the step of storing said decrypted application on said IC card.

3. The method of claim 1, wherein said encryption technique using said transfer key transfer key is symmetric.

4. The method of claim 3, wherein said symmetric technique is DES.

ANNEX A TO THE DESCRIPTION

5. The method of claim 1, wherein said IC card's public and private keys are provided using an asymmetric technique.

6. The method of claim 5, wherein said asymmetric technique is RSA.

7. The method of claim 1, wherein said key transformation unit further indicates the technique used to encrypt said at least a portion of said application.

8. The method of claim 1, further including the steps of enciphering a second portion of said application exclusive of said at least a portion of said application.

9. The method of claim 8, wherein said second portion is encrypted using a second encryption technique and said key transformation unit indicates said second encryption technique.

10. The method of claim 8, wherein said second portion is encrypted using a second key and said key transformation unit indicates said second key.

11. The method of claim 8, wherein said key transformation unit indicates the location of said second portion of said application.

ANNEX A TO THE DESCRIPTION

12. The method of claim 1, wherein said key transformation unit indicates the location of said at least a portion of said application.

13. The method of claim 1, wherein said key transformation unit indicates the number of encrypted portions of said application.

14. The method of claim 1, further including the steps of providing a public key and secret key set for an application provider; providing a public and secret key set for a certification authority; encrypting said application provider's public key using said certificate authorities' secret key to produce an application load certificate; further signing said encrypted application using said application provider's secret key to produce a signed application and transmitting said signed application and said application load certificate to said IC card.

15. The method of claim 14, further including the step of the IC card verifying said application load certificate with said certification authority's public key.

16. The method of claim 15, further including the steps of verifying the signed encrypted application using the application provider's public key from said decrypted application load certificate.

17. The method of claim 16, wherein said verified application signature is compared to sent encrypted application to determine if they are equivalent.

18. An IC card system comprising:

at least one IC card;

an application provider for providing an application to said at least one IC card;

a communications link coupled to said at least one IC card and said application provider;

a public key and secret key set generated for said IC card;

a transport key generated for use by said applications provider; and

an application, wherein at least a portion of said application is encrypted by said application provider using said transport key; said transport key is encrypted using said IC card's public key to form a key transformation unit; wherein said encrypted application and said key transformation unit are then transmitted to said IC card over said communications link; said transmitted key transformation unit is decrypted using said IC card's private key to recover said transport key; and said transmitted application is decrypted using said recovered transport key to recover said application.

19. The system of claim 18, wherein said recovered application is stored on said card.

ANNEX A TO THE DESCRIPTION

20. The system of claim 18, wherein said encryption technique using said transfer key transfer key is symmetric.

21. The system of claim 20, wherein said symmetric technique is DES.

22. The system of claim 18, wherein said IC card's public and private keys are provided using an asymmetric technique.

23. The system of claim 22, wherein said asymmetric technique is RSA.

24. The system of claim 18, wherein said key transformation unit further indicates the technique used to encrypt said at least a portion of said application.

25. The system of claim 18. further including the steps of enciphering a second portion of said application independently of said at least a portion of said application.

26. The system of claim 25, wherein said second portion is encrypted using a second encryption technique and said key transformation unit indicates said second encryption technique.

27. The system of claim 25, wherein said second portion is encrypted using a second key and said key transformation unit indicates said second key.

ANNEX A TO THE DESCRIPTION

28. The system of claim 25, wherein said key transformation unit indicates the location of said second portion of said application.

29. The system of claim 18, wherein said key transformation unit indicates the location of at least a portion of said application.

30. The system of claim 18, wherein said key transformation unit indicates the number of encrypted portions of said application.

31. The system of claim 18, further including a certification authority, wherein a public key and secret key set is provided for an application provider; a public and secret key set is provided for said certification authority; said certificate authority's secret key is used to sign said application provider's public key to produce an application load certificate; said application provider's secret key is used to further sign said encrypted application to produce a signed encrypted application and said signed encrypted application and said application load certificate is transmitted to said IC card.

32. The system of claim 31, wherein the IC card verifies said application load certificate with said certification authority's public key.

ANNEX A TO THE DESCRIPTION

33. The system of claim 32, wherein said IC card verifies the signed encrypted application using the application provider's public key from said verified application load certificate.

34. The system of claim 33, wherein said verified application signature is compared to said encrypted application to determine if they are equivalent.

35. A method for transmitting data in a secure manner from a first microprocessor based device to a second microprocessor based device, comprising the steps of:

encrypting at least a portion of said data at said first device using a transfer key;

encrypting said transfer key with a second key at said first device to form a key transformation unit;

transmitting said encrypted data and said key transformation unit to said second device;

decrypting said key transformation unit at said second device to recover said transfer key; and

decrypting said encrypted data using said recovered transfer key.

36. The method of claim 35, further including the step of storing said decrypted data in said second device.

ANNEX A TO THE DESCRIPTION

37. The method of claim 35, wherein said second key is from a public key and private key set used in asymmetric encryption.

38. The method of claim 35, wherein said key transformation unit further indicates the technique used to encrypt said at least a portion of said application.

39. The method of claim 35, further including the steps of enciphering a second portion of said application independently of said at least a portion of said application.

40. The method of claim 39, wherein said second portion is encrypted using a second encryption technique and said key transformation unit indicates said second encryption technique.

41. The method of claim 39, wherein said second portion is encrypted using a second key and said key transformation unit indicates said second key.

42. The method of claim 39, wherein said key transformation unit indicates the location of said second portion of said application.

43. The method of claim 35, wherein said key transformation unit indicates the location of said at least a portion of said application.

ANNEX A TO THE DESCRIPTION

44. The method of claim 35, further including the steps of providing a public key and secret key set for an application provider; providing a public and secret key set for a certification authority; signing said application provider's public key using said certificate authority's secret key to produce an application load certificate; further signing said encrypted application using said application provider's secret key to produce a signed encrypted application and transmitting said signed application and said application load certificate to said IC card.

45. A method for processing a data transmission comprising the steps of:

receiving said data transmission comprising an application encrypted with a first key and a key transformation unit encrypted with a second key, wherein said key transformation unit comprises said first key;

decrypting said key transformation unit to recover said first key;

decrypting said encrypted application using said first key; and

storing said decrypted application.

46. The method of claim 45, wherein said second key is from a public key and private key set used in asymmetric encryption.

47. The method of claim 45, wherein said key transformation unit further indicates the technique used to encrypt said at least a portion of said application.

ANNEX A TO THE DESCRIPTION

48. The method of claim 45, further including the steps of enciphering a second portion of said application independently of said at least a portion of said application.

49. The method of claim 48, wherein said second portion is encrypted using a second encryption technique and said key transformation unit indicates said second encryption technique.

50. The method of claim 48, wherein said second portion is encrypted using a second key and said key transformation unit indicates said second key.

51. The method of claim 48, wherein said key transformation unit indicates the location of said second portion of said application.

52. The method of claim 45, wherein said key transformation unit indicates the location of said at least a portion of said application.

53. The method of claim 45, further including the steps of providing a public key and secret key set for an application provider; providing a public and secret key set for a certification authority; signing said application provider's public key using said certificate authorities' secret key to produce an application load certificate; further encrypting said encrypted application using said application provider's secret key to produce a signed encrypted application and transmitting

ANNEX A TO THE DESCRIPTION

said signed application and said application load certificate to said IC card.

54. The method of claim 53, further including the step of the IC card verifying said application load certificate with said certification authority's public key.

55. The method of claim 54, further including the steps of verifying the signed encrypted application using the application provider's public key from said verified application load certificate.

56. The method of claim 55, wherein said verified application signature is compared to said encrypted application to determine if they are equivalent.

57. An apparatus for processing a data transmission comprising the steps of:

means for receiving said data transmission comprising an application encrypted with a first key and a key transformation unit encrypted with a second key, wherein said key transformation unit comprises said first key;

means for decrypting said key transformation unit to recover said first key;

means for decrypting said encrypted application using said first key; and

means for storing said decrypted application.

ANNEX A TO THE DESCRIPTION

58. The apparatus of claim 57, wherein said second key is from a public key and private key set used in asymmetric encryption.

59. The apparatus of claim 57, wherein said key transformation unit further indicates the technique used to encrypt said at least a portion of said application.

60. The apparatus of claim 57, further including means for enciphering a second portion of said application exclusive of said at least a portion of said application.

61. The apparatus of claim 60, wherein said second portion is encrypted using a second encryption technique and said key transformation unit indicates said second encryption technique.

62. The apparatus of claim 60, wherein said second portion is encrypted using a second key and said key transformation unit indicates said second key.

63. The apparatus of claim 60, wherein said key transformation unit indicates the location of said second portion of said application.

64. The apparatus of claim 57, wherein said key transformation unit indicates the location of said at least a portion of said application.

ANNEX A TO THE DESCRIPTION

65. The apparatus of claim 60, further including means for verifying an application load certificate with said certification authority's public key.

66. The apparatus of claim 65, further including means for verifying the signed encrypted application using an application provider's public key located in said verified application load certificate.

67. The apparatus of claim 66, wherein said verified application signature is compared to the said encrypted application to determine if they are equivalent.

ANNEX A TO THE DESCRIPTION

ABSTRACT OF THE DISCLOSURE

A multi-application IC card system and method is disclosed providing a secure data transmission technique. The method is used, for example, to load an application from an application provider, which could be remote, to an IC card. At least a portion of the application is encrypted using a transfer key. The transfer key is then encrypted using the public key of a public/secret key pair of the intended IC card to form a key transformation unit. The encrypted application and key transformation unit are then sent to the IC card and the IC card decrypts the key transformation unit using its secret key. The transfer key is then recovered and used to decrypt the encrypted application. The application can then by stored on the IC card and accessed by the card user.

WE CLAIM:

1. A method for transporting data onto an integrated circuit card by using an individualized key set for said card, comprising the steps of:

storing a private key and public key pair unique to said integrated circuit card in said memory located on said integrated circuit card;

retrieving said stored public key from said integrated circuit card;

encrypting at least a portion of said data to be transported onto said card, using said retrieved public key;

transmitting said encrypted data to said integrated circuit card; and

decrypting said encrypted data using said integrated circuit card's private key to recover said transported data.

2. The method of claim 1, further including the step of storing said decrypted data on said integrated circuit card.

3. The method of claim 1 or claim 2, wherein a certification authority digitally signs said integrated circuit card's public key to produce a public key certificate unique to said card and stored thereon, and wherein said public key certificate is verified prior to said transmitting step.

4. The method of claim 3, wherein said public key certificate is verified with said certification authority's stored public key prior to said transmitting steps.

5. The method of claim 3 or 4, wherein said retrieved public key certificate is recovered and compared with said stored public key.

6. The method of any preceding claim, wherein said integrated circuit card's public and private keys are provided using an asymmetric technique.

7. The method of claim 6, wherein said asymmetric technique is RSA.

8. A method performed by an integrated circuit card for processing incoming data transmission to said integrated circuit card by using an individualized key set for the card, comprising the steps of:

receiving said data transmission comprising data encrypted with a public key stored on said integrated circuit card, said public key forming part of said individualized key set;

retrieving a unique private key for said integrated circuit card which is part of said individualized key set; and

decrypting said encrypted data with said unique private key to recover said data.

9. The method of claim 8, further including the step of storing said decrypted data on said integrated circuit card.

10. The method of claim 8 or 9, wherein said individualized key set is generated by asymmetric encryption.

11. The method of any of claims 8 to 10, wherein a certification authority digitally signs said integrated circuit card's public key to produce a public key certificate unique to said card and stored thereon. and wherein said public key certificate is verified prior to said transmitting step.

12. The method of claim 11, wherein said public key certificate is retrieved prior to said transmitting steps.

13. The method of claim 11 or 12, wherein said retrieved public key certificate is verified using said certification authority's stored public key.

14. An apparatus located on an integrated circuit card by using an individualized key set for said card for processing an incoming secure data transmission comprising:

means for receiving said data transmission comprising data encrypted with a public key stored on said integrated circuit card, said public key forming part of said individualized key set;

means for retrieving a unique public key for said integrated circuit card which is part of said individualized key set; and

means for decrypting said encrypted data with said unique private key to recover said data.

15. The apparatus of claim 14, further comprising means for storing said data on said integrated circuit card.

16. The apparatus of claim 14 or 15, further including means for retrieving a public key certificate which is generated by a certificate authority digitally signing said unique public key.

17. The apparatus of claim 16, further including means for transmitting said public key certificate prior to said receiving means receiving.

18. The apparatus of claim 16 or 17, wherein said transmitted public key certificate is verified using said certification authority's stored public key.

19. A method of transporting data onto an integrated circuit card by using an individualized key set for the card, comprising the steps of:

providing a first unique private and public key pair for a certification authority;

storing a second unique private and public key pair which

form said individualized key set for said integrated circuit card in a memory located on said integrated circuit card;

encrypting said second public key with said first certification authority's private key to form a public key certificate;

storing said public key certificate on said integrated circuit card;

retrieving said stored public key certificate from said integrated circuit card;

verifying said public key certificate with said first public key to ensure that said public key certificate is valid;

encrypting at least a portion of said data using said retrieved second public key;

transporting said encrypted data to said integrated circuit card; and

decrypting said encrypted data using said second private key to retrieve said data.

20. The method of claim 19, wherein said data comprises an application.

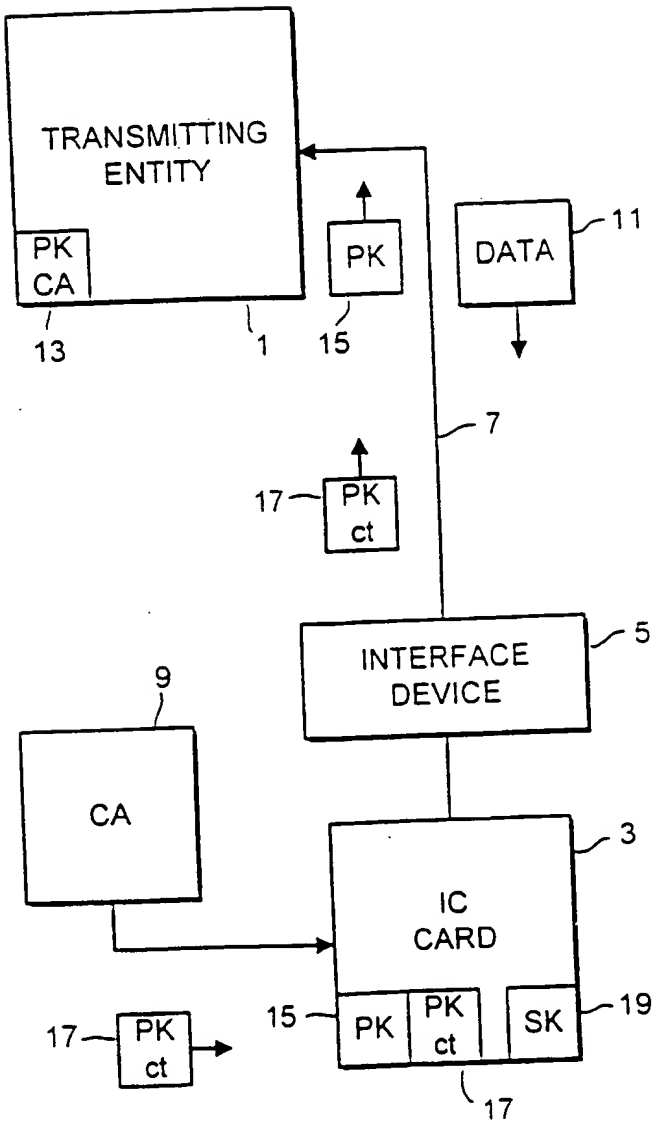

TRANSMITTING ENTITY
PK CA
13
1
PK
15
DATA
11
7
17
PK ct
INTERFACE DEVICE
5
9
CA
IC CARD
3
15
PK
PK ct
SK
19
17
PK ct
17

FIG. 1A

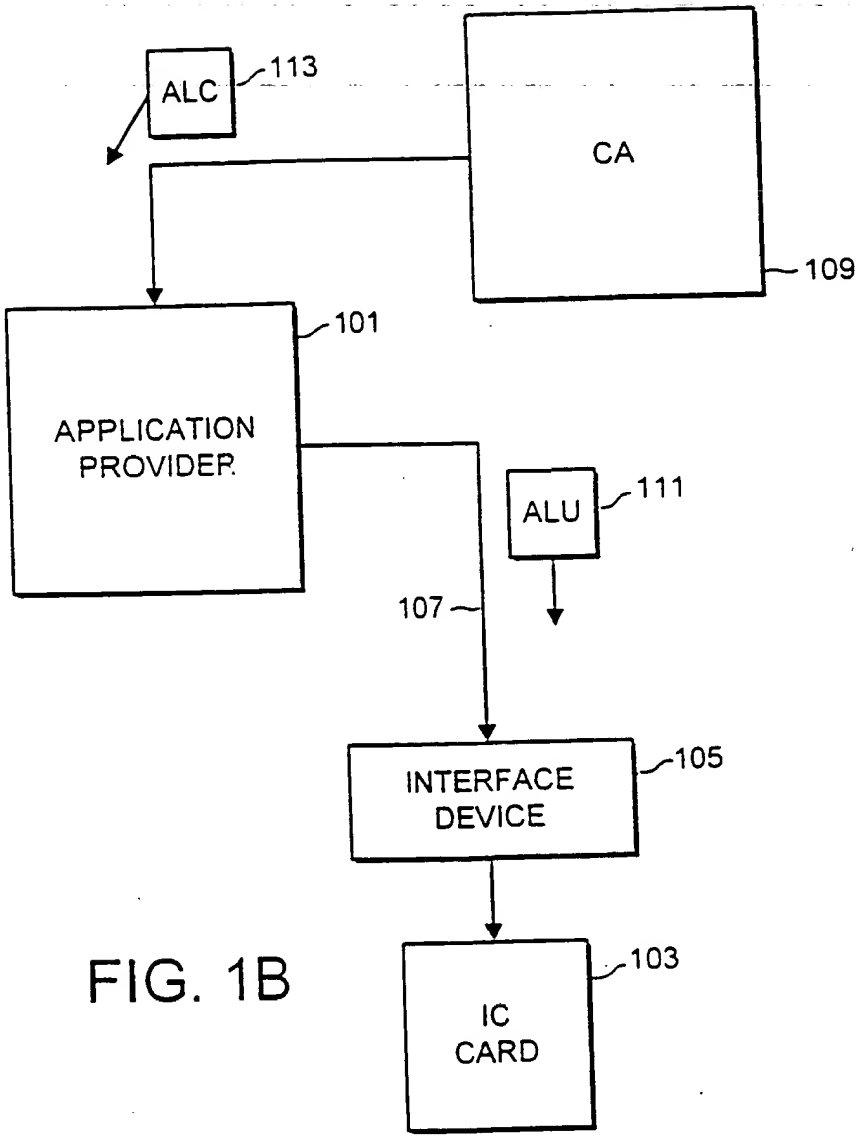

ALC
113
CA
109
101
APPLICATION PROVIDER
ALU
111
107
INTERFACE DEVICE
105
IC CARD
103

FIG. 1B

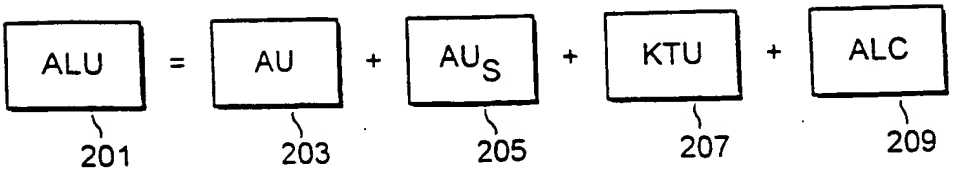

ALU
=
AU
+
AUS
+
KTU
+
ALC
201
203
205
207
209

FIG. 2

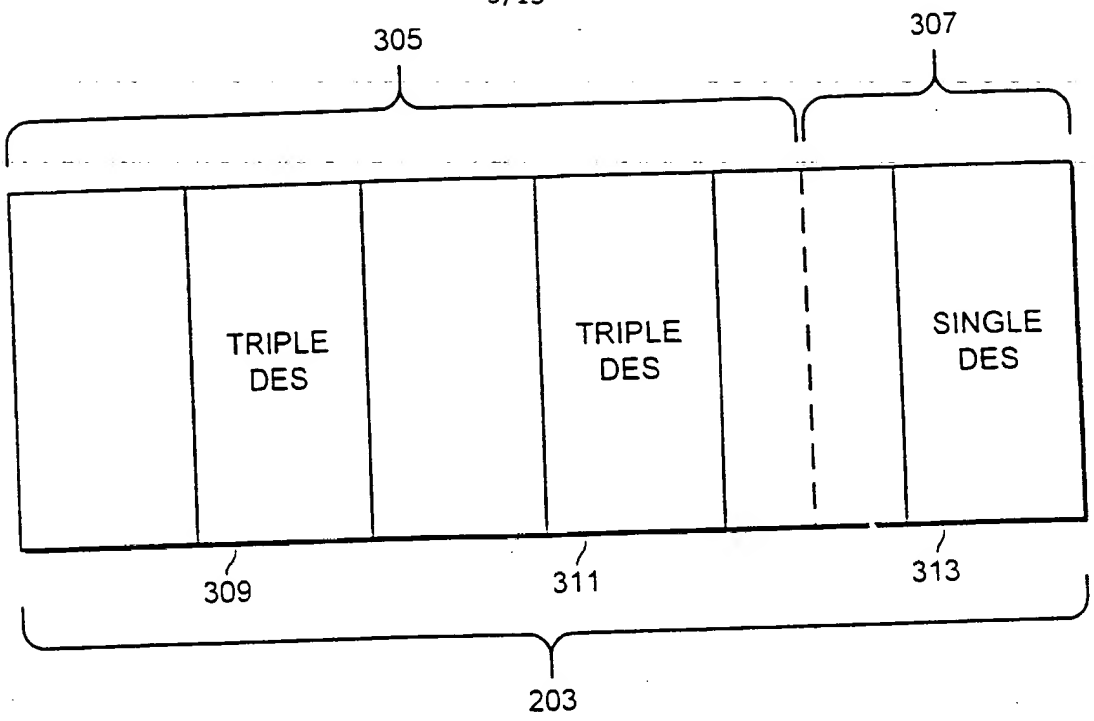

305
307
TRIPLE DES
TRIPLE DES
SINGLE DES
309
311
313
203

FIG. 3

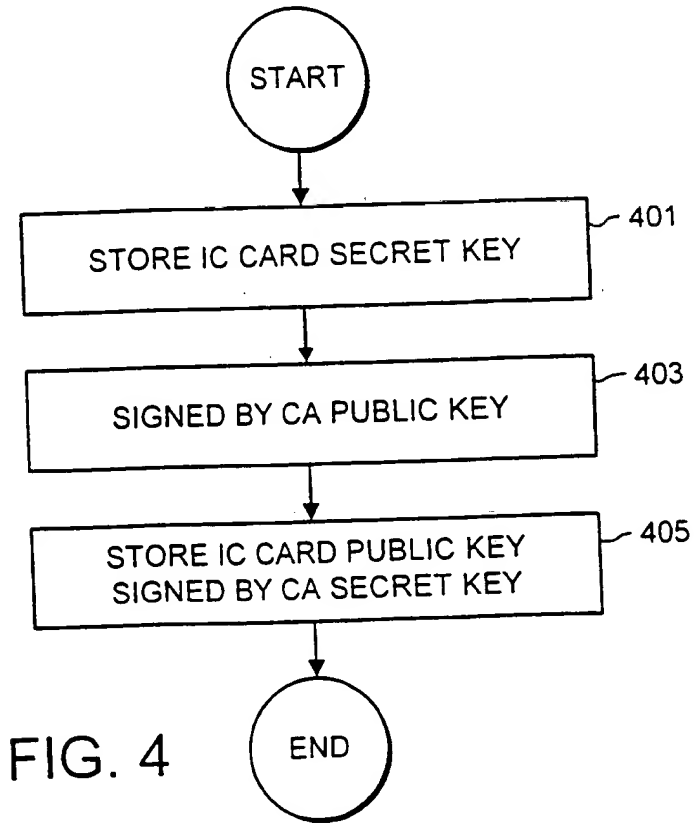

START
STORE IC CARD SECRET KEY
401
SIGNED BY CA PUBLIC KEY
403
STORE IC CARD PUBLIC KEY SIGNED BY CA SECRET KEY
405
END

FIG. 4

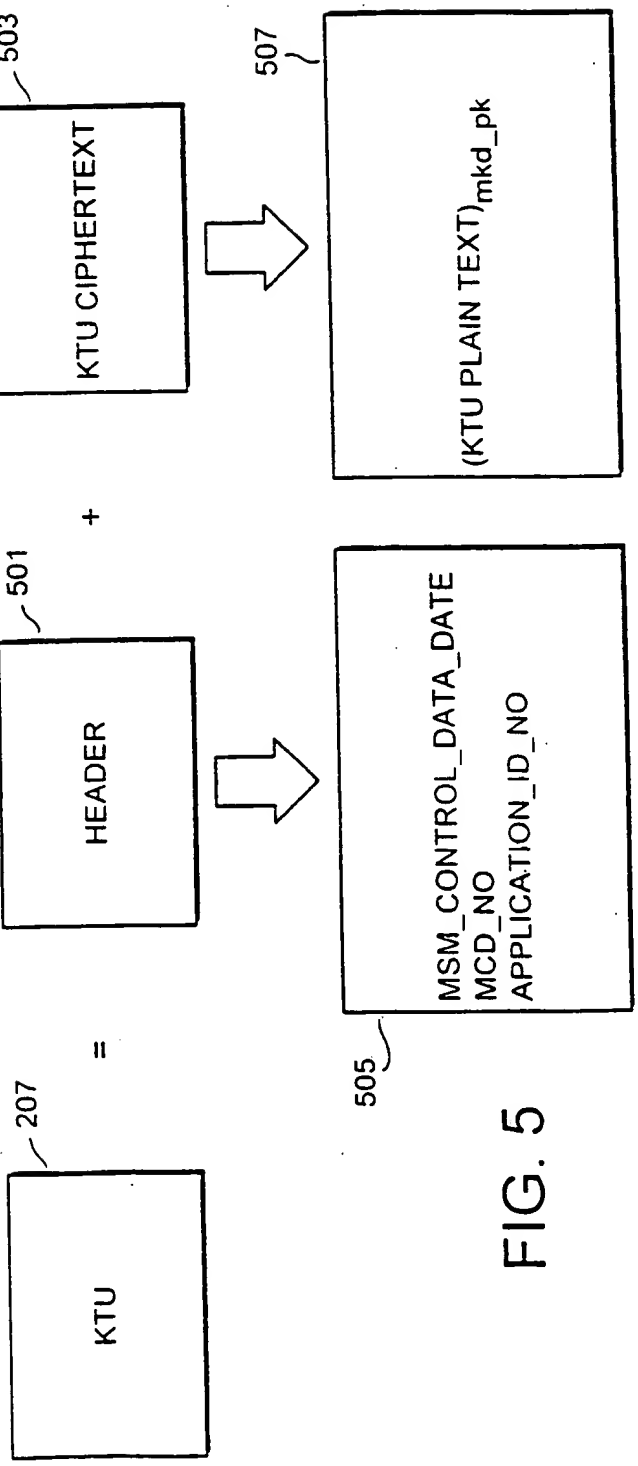

KTU
207
=
HEADER
501
+
KTU CIPHERTEXT
503
MSM_CONTROL_DATA_DATE
MCD_NO
APPLICATION_ID_NO
505
(KTU PLAIN TEXT)mkd_pk
507

FIG. 5

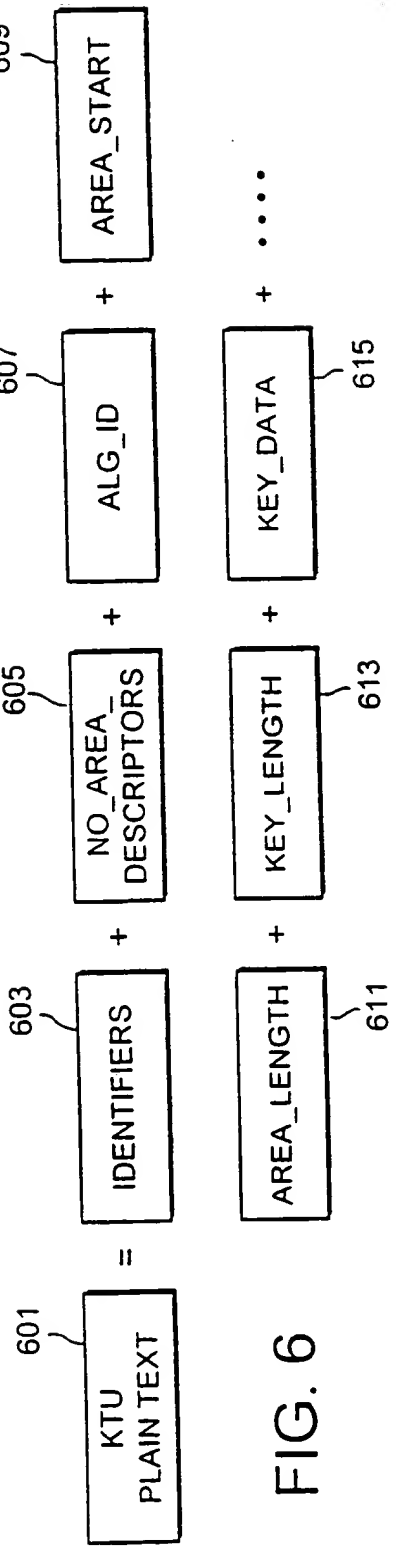

KTU PLAIN TEXT
601
=
IDENTIFIERS
603
+
NO_AREA_DESCRIPTORS
605
+
ALG_ID
607
+
AREA_START
609
AREA_LENGTH
611
+
KEY_LENGTH
613
+
KEY_DATA
615
+ ....

FIG. 6

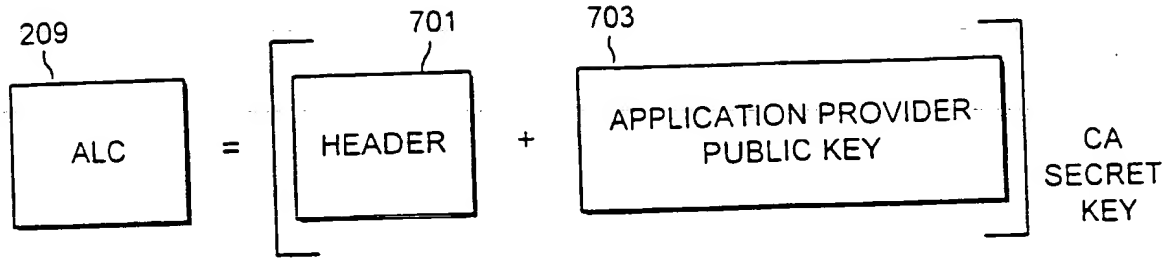


FIG. 7

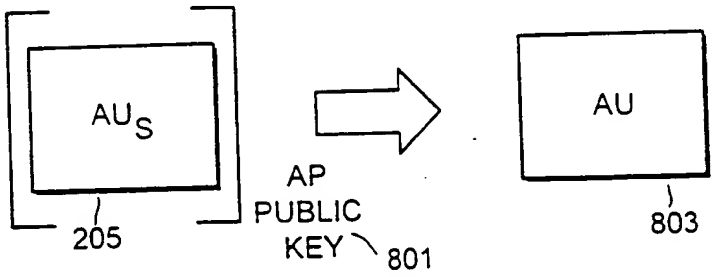


FIG. 8

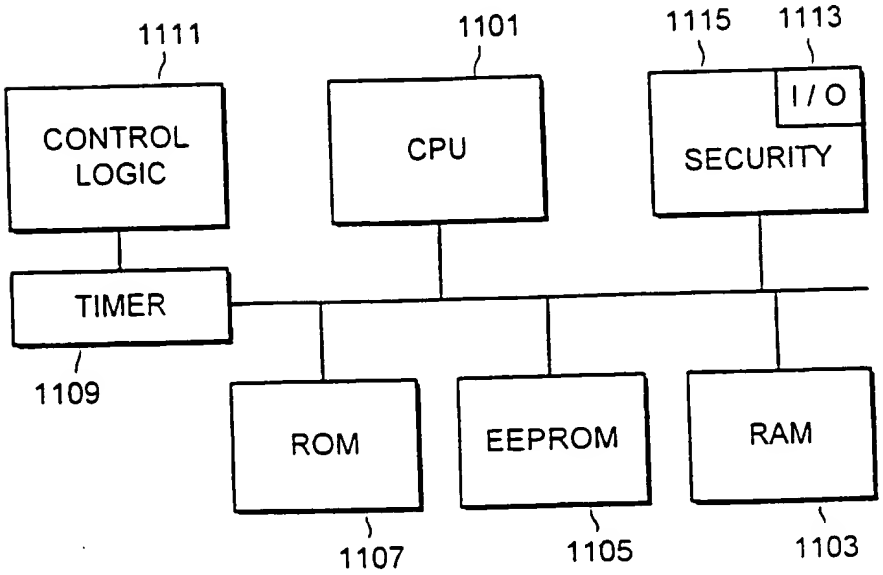


FIG. 11

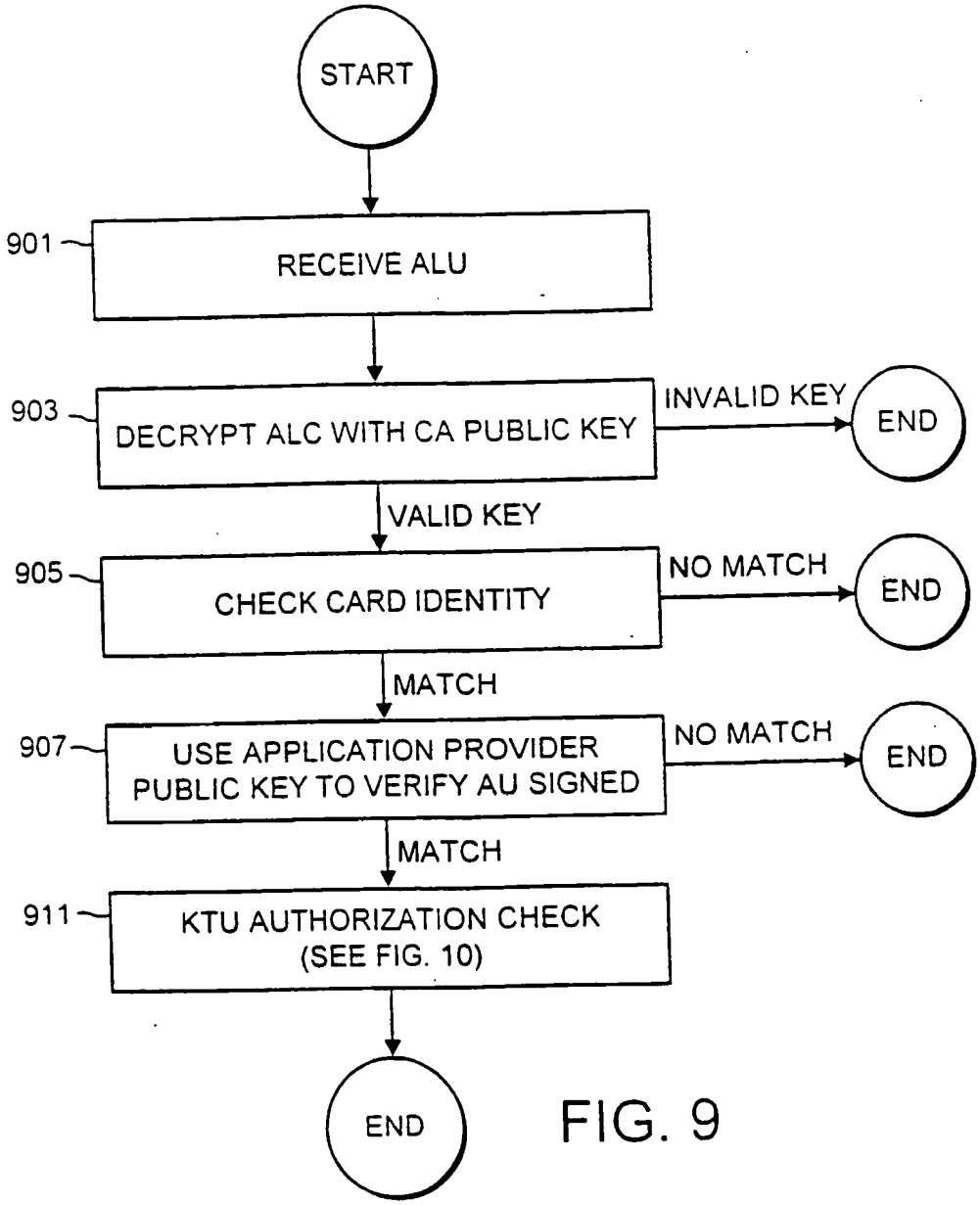

START
901
RECEIVE ALU
903
DECRYPT ALC WITH CA PUBLIC KEY
INVALID KEY
END
VALID KEY
905
CHECK CARD IDENTITY
NO MATCH
END
MATCH
907
USE APPLICATION PROVIDER PUBLIC KEY TO VERIFY AU SIGNED
NO MATCH
END
MATCH
911
KTU AUTHORIZATION CHECK (SEE FIG. 10)
END

FIG. 9

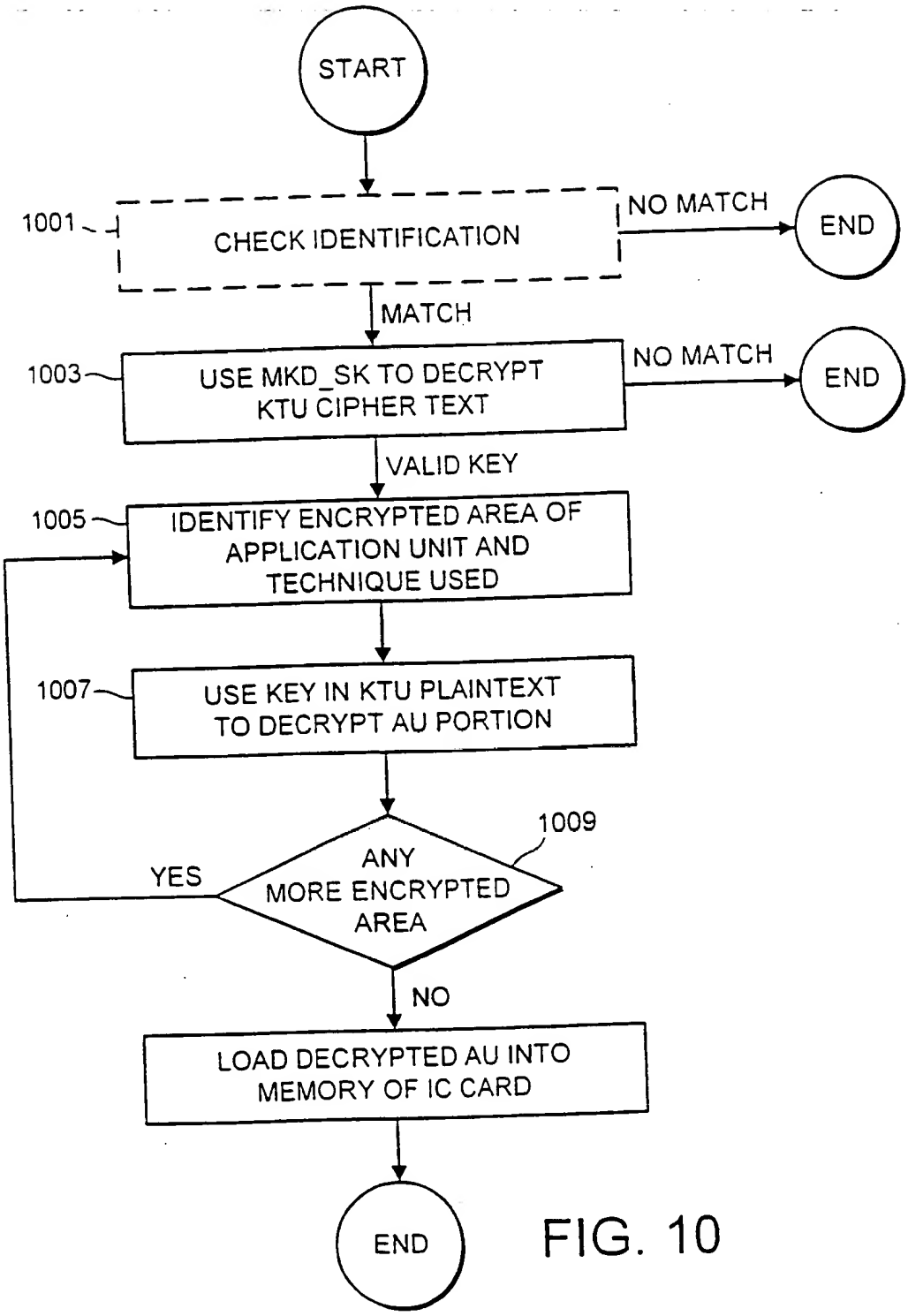

START
1001
CHECK IDENTIFICATION
NO MATCH
END
MATCH
1003
USE MKD_SK TO DECRYPT KTU CIPHER TEXT
NO MATCH
END
VALID KEY
1005
IDENTIFY ENCRYPTED AREA OF APPLICATION UNIT AND TECHNIQUE USED
1007
USE KEY IN KTU PLAINTEXT TO DECRYPT AU PORTION
1009
ANY MORE ENCRYPTED AREA
YES
NO
LOAD DECRYPTED AU INTO MEMORY OF IC CARD
END

FIG. 10

ANNEX A TO THE DRAWINGS

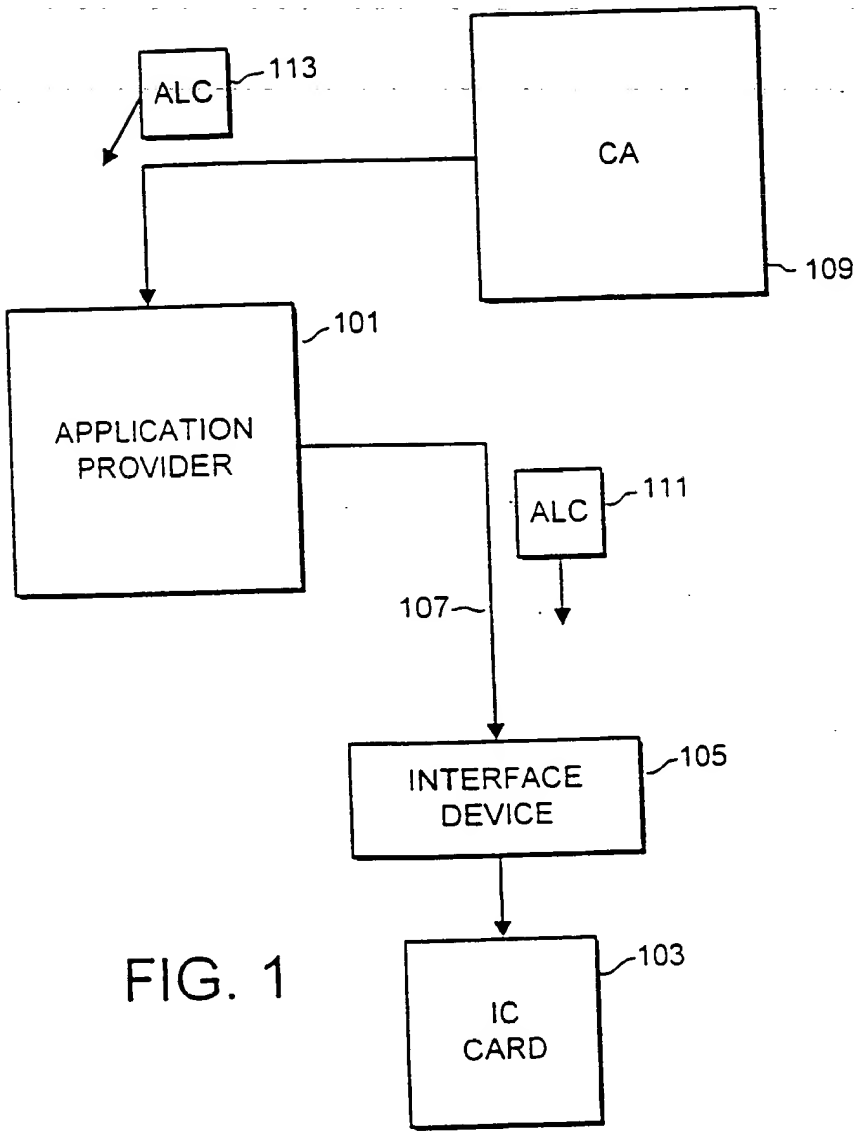


FIG. 1

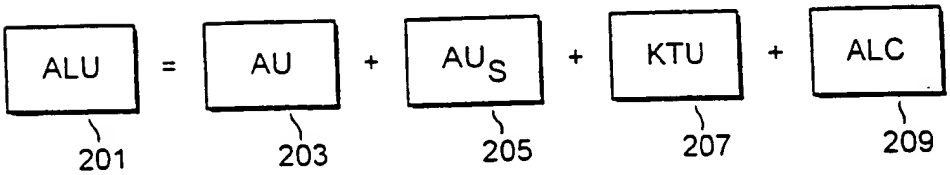


FIG. 2

ANNEX A TO THE DRAWINGS

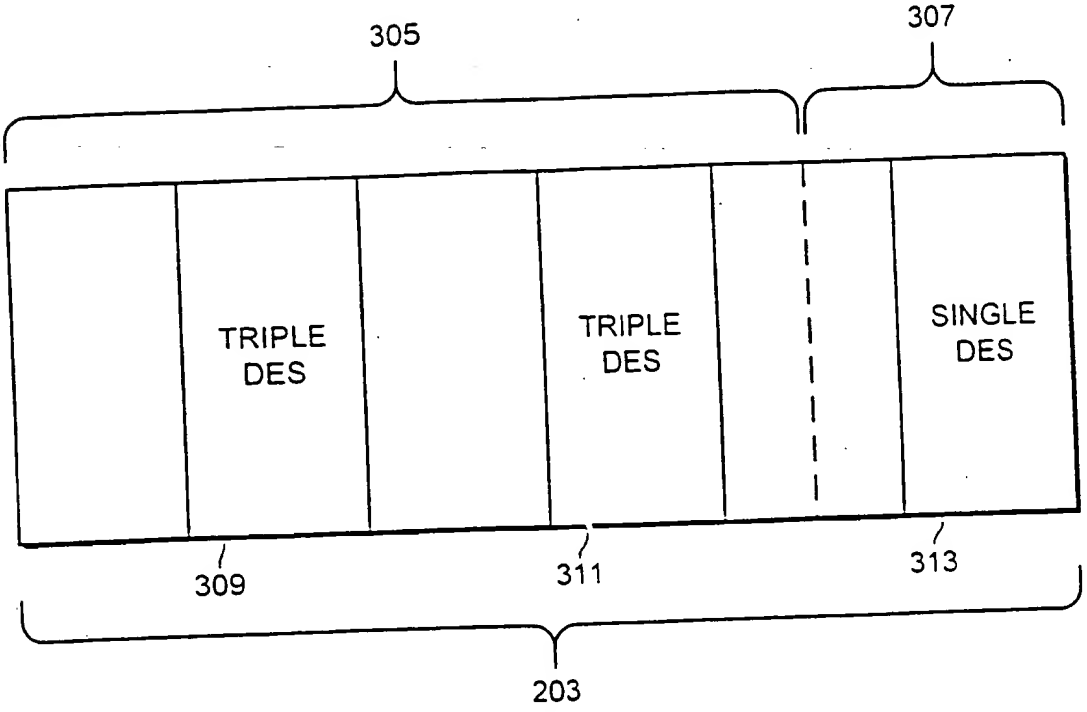


FIG. 3

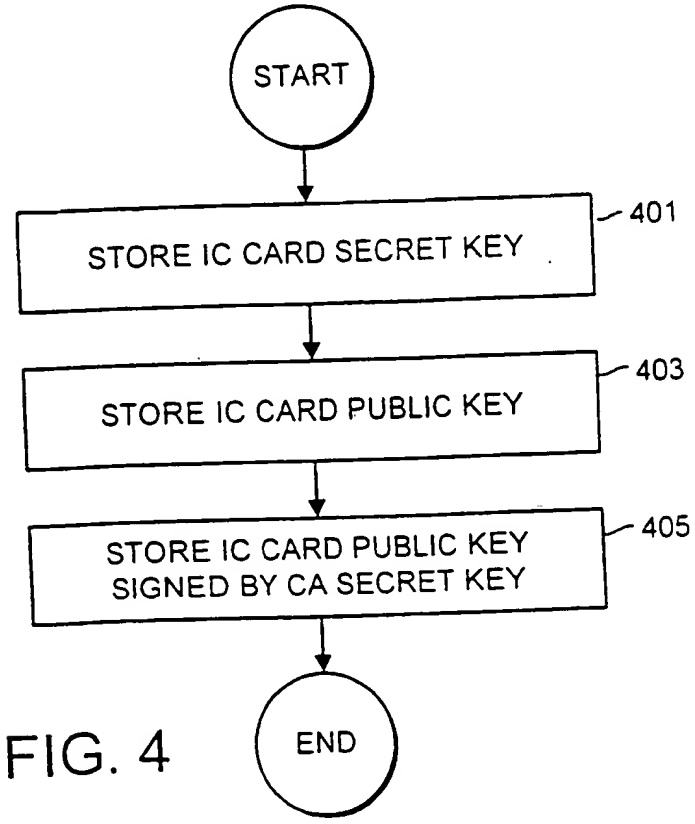


FIG. 4

ANNEX A TO THE DRAWINGS

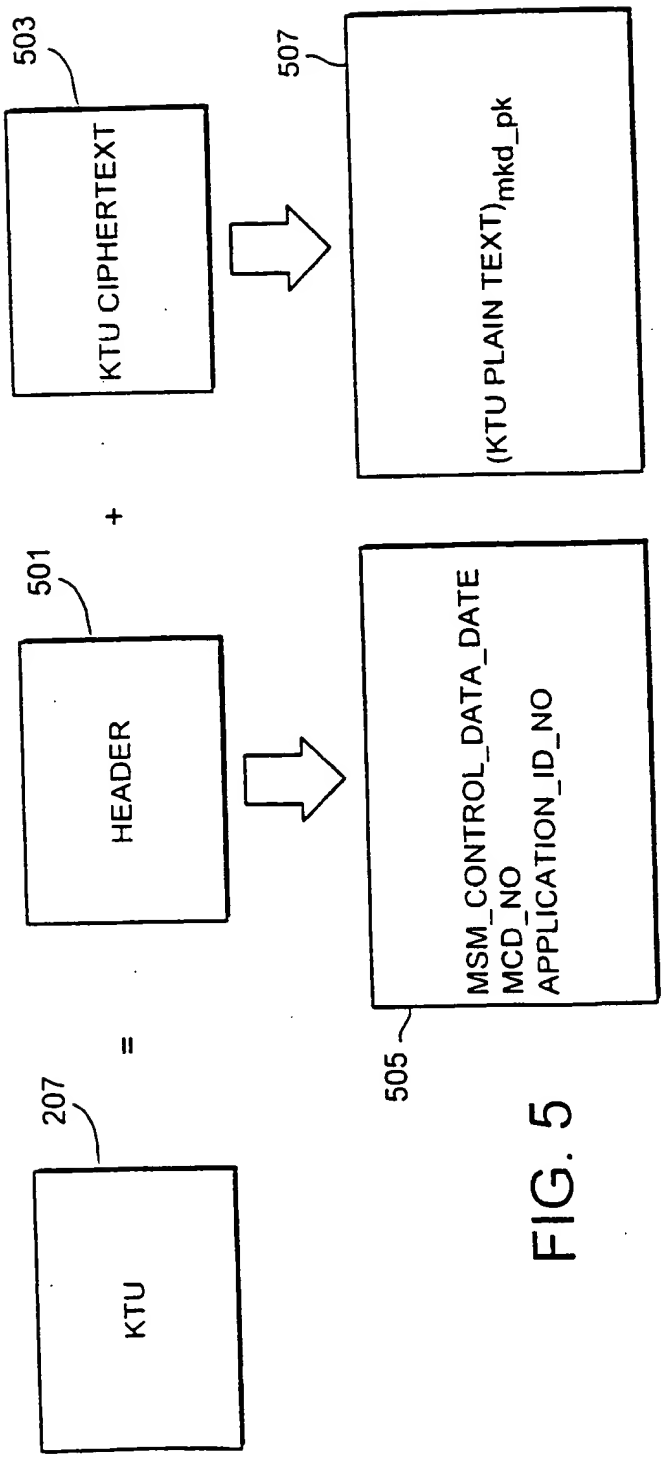


FIG. 5

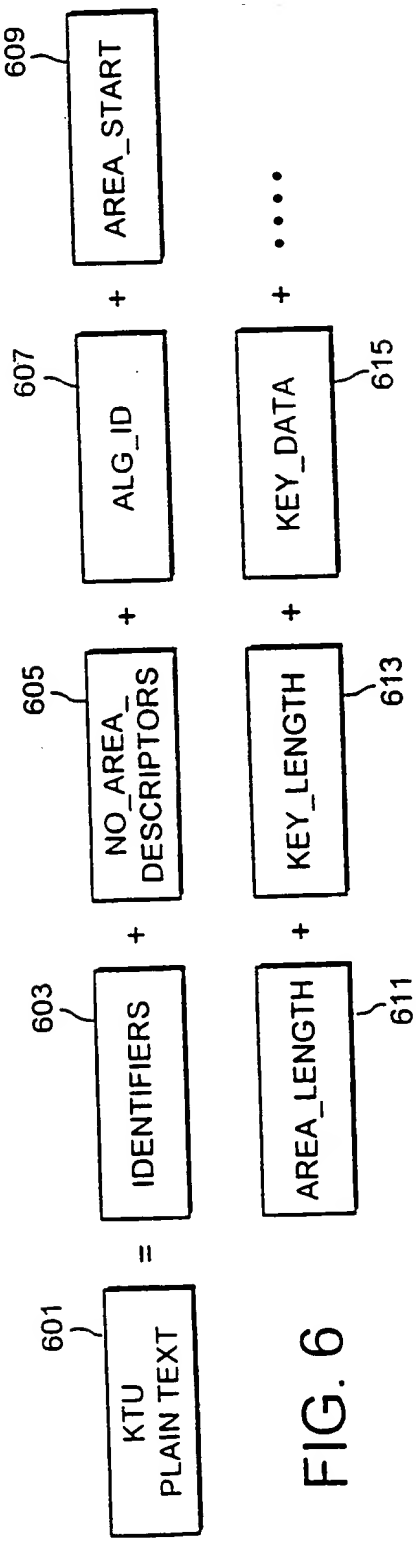


FIG. 6

ANNEX A TO THE DRAWINGS

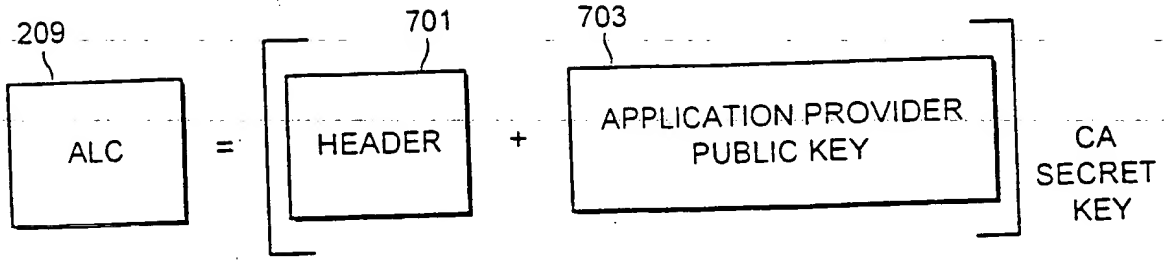


FIG. 7

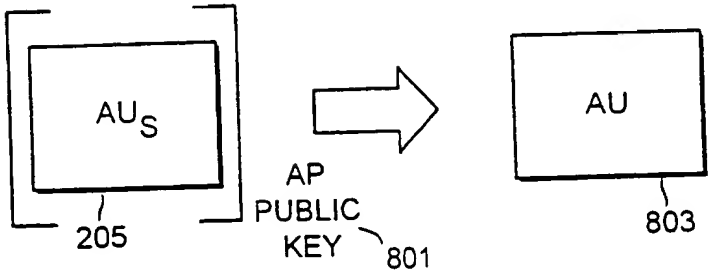


FIG. 8

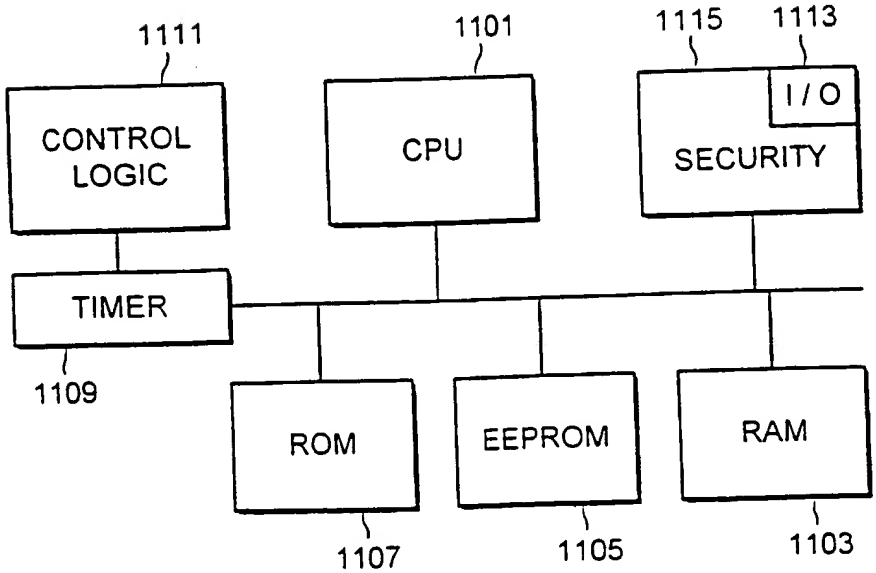


FIG. 11

ANNEX A TO THE DRAWINGS

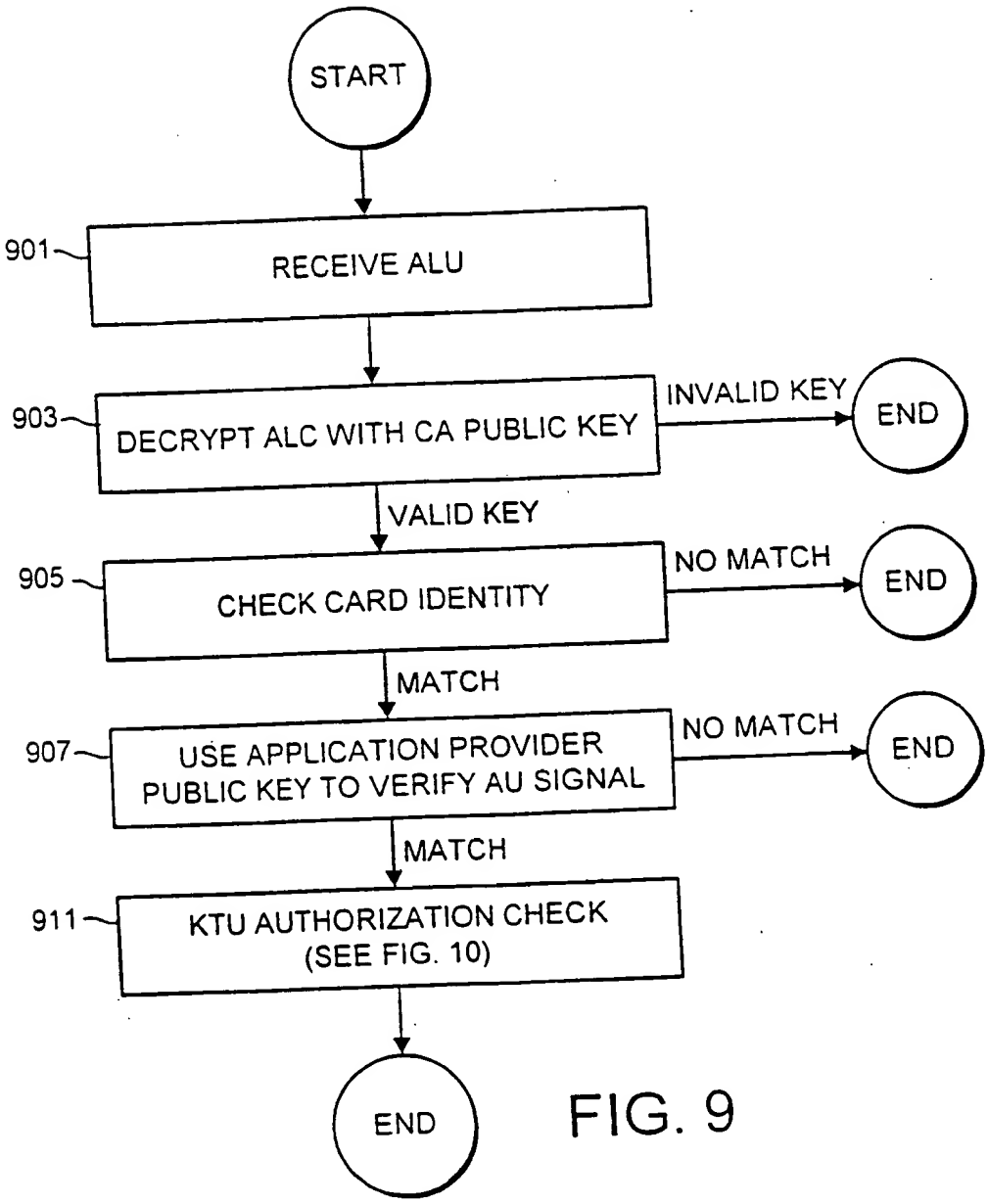


FIG. 9

ANNEX A TO THE DRAWINGS

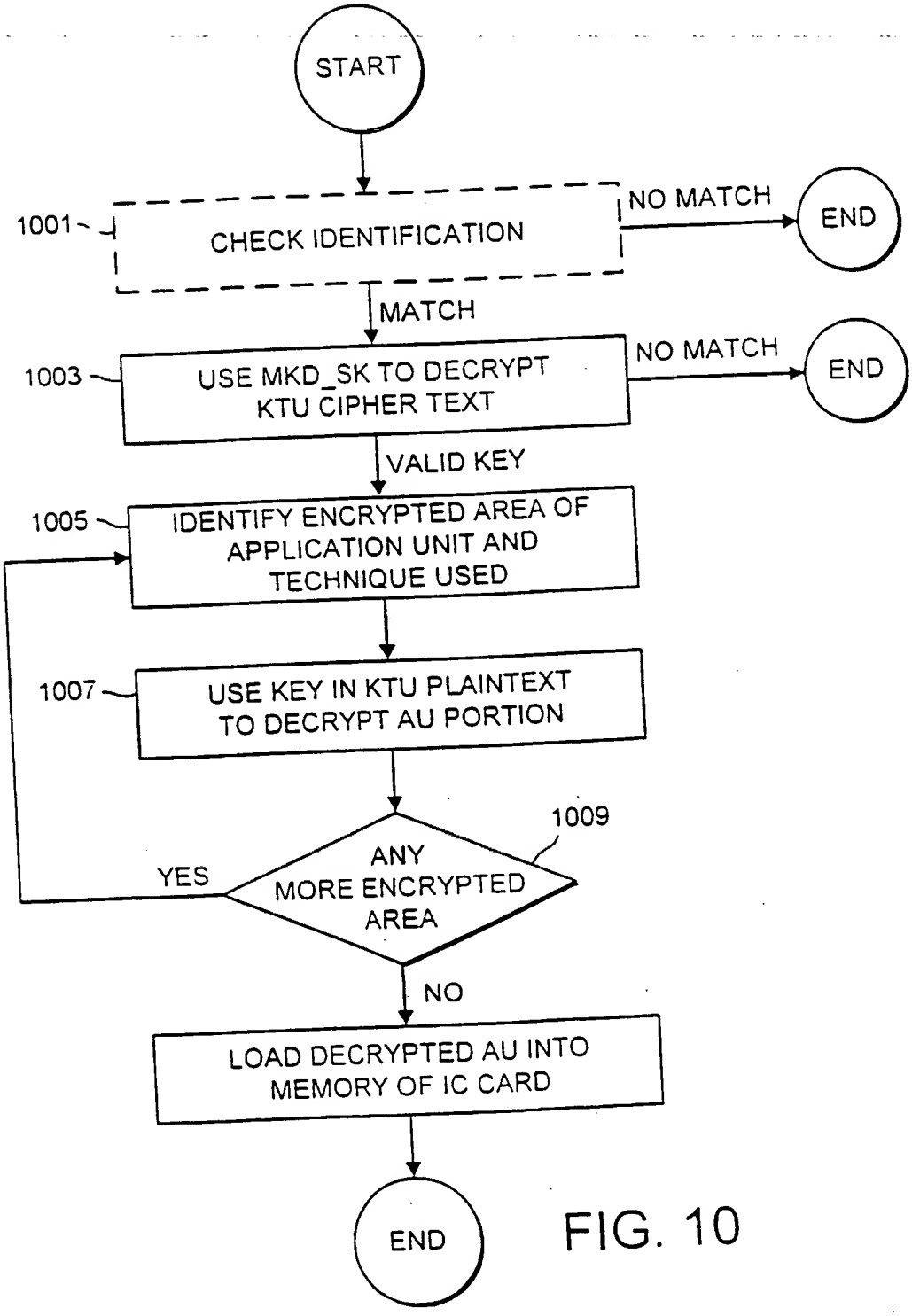


FIG. 10

PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : G07F 7/10, H04L 9/32

A3

(11) International Publication Number: **WO 98/52163**

(43) International Publication Date: 19 November 1998 (19.11.98)

(21) International Application Number: PCT/GB98/01405

(22) International Filing Date: 14 May 1998 (14.05.98)

(30) Priority Data:

60/046,514	15 May 1997 (15.05.97)	US
09/075,973	11 May 1998 (11.05.98)	US

(71) Applicant: MONDEX INTERNATIONAL LIMITED [GB/GB]; 47–53 Cannon Street, London EC4M 5SQ (GB).

(72) Inventors: RICHARDS, Timothy, Philip; 32 Craig Mount, Radlett, Herts. WD7 7LW (GB). EVERETT, David, Barrington; 31 Ashdown Avenue, Saltdean, Brighton, Easr Sussex BN2 8AH (GB). VINER, John, Charles; Hydes, Woodlands Lane, Windlesham (GB).

(74) Agent: POTTER, Julian, Mark; D. Young & Co., 21 New Fetter Lane, London EC4A 1DA (GB).

(81) Designated States: AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GE, GH, GM, GW, HU, ID, IL, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, UZ, VN, YU, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG).

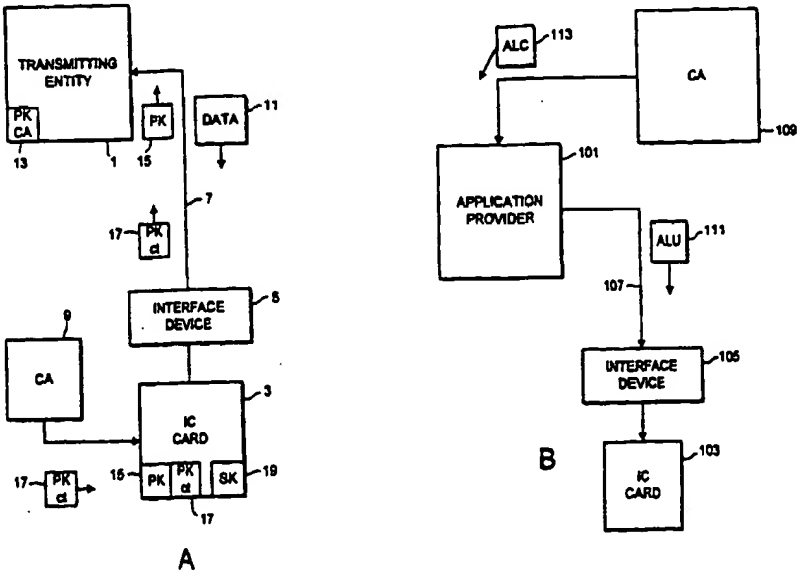
Published

With international search report.

Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(88) Date of publication of the international search report: 17 June 1999 (17.06.99)

(54) Title: IC CARD TRANSPORTATION KEY SET



(57) Abstract

Method and apparatus for securely transporting data onto an IC card. The method is used, for example, to transport data, including application programs, in a secure manner from a source located outside the IC card. At least a portion of the data is encrypted using the public key of a public/secret key pair of the intended IC card unit. The encrypted data is then sent to the IC card and the IC card verifies the key transformation unit using its unique secret key. The data can then be stored on the IC card. A copy of the public key signed by a certification authority can be used to verify that the card is authorized to be part of the overall authorized system.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania
AM	Armenia
AT	Austria
AU	Australia
AZ	Azerbaijan
BA	Bosnia and Herzegovina
BB	Barbados
BE	Belgium
BF	Burkina Faso
BG	Bulgaria
BJ	Benin
BR	Brazil
BY	Belarus
CA	Canada
CF	Central African Republic
CG	Congo
CH	Switzerland
CI	Côte d'Ivoire
CM	Cameroon
CN	China
CU	Cuba
CZ	Czech Republic
DE	Germany
DK	Denmark
EE	Estonia
ES	Spain
FI	Finland
FR	France
GA	Gabon
GB	United Kingdom
GE	Georgia
GH	Ghana
GN	Guinea
GR	Greece
HU	Hungary
IE	Ireland
IL	Israel
IS	Iceland
IT	Italy
JP	Japan
KE	Kenya
KG	Kyrgyzstan
KP	Democratic People's Republic of Korea
KR	Republic of Korea
KZ	Kazakstan
LC	Saint Lucia
LI	Liechtenstein
LK	Sri Lanka
LR	Liberia
LS	Lesotho
LT	Lithuania
LU	Luxembourg
LV	Latvia
MC	Monaco
MD	Republic of Moldova
MG	Madagascar
MK	The former Yugoslav Republic of Macedonia
ML	Mali
MN	Mongolia
MR	Mauritania
MW	Malawi
MX	Mexico
NE	Niger
NL	Netherlands
NO	Norway
NZ	New Zealand
PL	Poland
PT	Portugal
RO	Romania
RU	Russian Federation
SD	Sudan
SE	Sweden
SG	Singapore
SI	Slovenia
SK	Slovakia
SN	Senegal
SZ	Swaziland
TD	Chad
TG	Togo
TJ	Tajikistan
TM	Turkmenistan
TR	Turkey
TT	Trinidad and Tobago
UA	Ukraine
UG	Uganda
US	United States of America
UZ	Uzbekistan
VN	Viet Nam
YU	Yugoslavia
ZW	Zimbabwe

INTERNATIONAL SEARCH REPORT

Internat ıl Application No
PCT/GB 98/01405

A. CLASSIFICATION OF SUBJECT MATTER
IPC 6 G07F7/10 H04L9/32

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 6 G07F H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	FR 2 536 928 A (ÉTAT FRANCAIS) 1 June 1984 see abstract; claims; figures see page 7, line 5 - line 25 see page 11, line 18 - page 12, line 31 ---	1,2, 6-10,14, 15,19
Y	WO 91 16691 A (JONHIG) 31 October 1991	1,2, 6-10,14, 15,19
A	see abstract; claims; figure 3 see page 13, line 13 - page 14, line 30 --- -/--	3-5, 11-13, 16-18

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
27 April 1999	28. 04. 1999

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	David, J

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

Internati Application No
PCT/GB 98/01405

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	EP 0 707 290 A (CP8 TRANSAC) 17 April 1996 see abstract; claims; figures see column 8, line 1 - line 52 ---	1,2,6, 8-10,14, 15,19,20
A	EP 0 588 339 A (NIPPON TELEGRAPH AND TELEPHONE) 23 March 1994 -----	

INTERNATIONAL SEARCH REPORT

Internat' Application No
PCT/GB 98/01405

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
FR 2536928 A	01-06-1984	NONE	
WO 9116691 A	31-10-1991	AT 127949 T	15-09-1995
		AU 653721 B	13-10-1994
		AU 7664491 A	11-11-1991
		CA 2058982 A,C	13-10-1991
		CN 1057535 A,B	01-01-1992
		DE 69112975 D	19-10-1995
		DE 69112975 T	29-02-1996
		DK 479982 T	13-11-1995
		EP 0479982 A	15-04-1992
		ES 2034929 T	16-11-1995
		GR 92300099 T	16-03-1993
		GR 3017457 T	31-12-1995
		HK 175596 A	27-09-1996
		KR 145331 B	17-08-1998
		NO 303198 B	08-06-1998
		PL 169723 B	30-08-1996
		US 5623547 A	22-04-1997
		US 5778067 A	07-07-1998
EP 0707290 A	17-04-1996	FR 2725537 A	12-04-1996
		AU 690324 B	23-04-1998
		AU 3318795 A	16-05-1996
		BR 9504355 A	08-10-1996
		CA 2160223 A	12-04-1996
		JP 8212066 A	20-08-1996
		NO 954028 A	12-04-1996
		US 5825875 A	20-10-1998
EP 0588339 A	23-03-1994	JP 6103425 A	15-04-1994
		JP 6103426 A	15-04-1994
		JP 6162289 A	10-06-1994
		JP 6162287 A	10-06-1994
		JP 6161354 A	07-06-1994
		DE 69322463 D	21-01-1999
		EP 0856821 A	05-08-1998
		EP 0856822 A	05-08-1998
		US 5396558 A	07-03-1995
		US 5446796 A	29-08-1995
		US 5502765 A	26-03-1996